Asian Journal of Information Technology 5 (4) : 388-391, 2006

Improved Control Vector Encryption/Decryption for Session-Key Distribution

[1]M. Ismail Jabiullah, [2]Abdullah Al-Shamim, [2]A.N.M. Khaleqdad Khan and [3]M. Lutfar Rahman
[1]Department of Computer Science and Engineering, Research Student,
Dhaka University and Associate Professor, Institute of Science and Technology
[2]Institute of Science and Technology, Dhaka, Bangladesh
[3]Department of Computer Science and Engineering University of Dhaka, Dhaka, Bangladesh

Abstract: Symmetric-key encryption/decryption process plays the important role in secure electronic communications. To work with symmetric-key cryptography, the communicating parties must share the same private key and that the key must be protected from access by others. Frequent key changes are usually desirable to limit the amount of data compromised if an intruder learns the key. The session-key on which the communication between the end-users is encrypted is used for the duration of a logical connection such as a frame relay connection or a transport connection and then discarded. Each session-key is obtained from the Key Distribution Center (KDC) over the same networking facilities used for end-user communication. The control vector is cryptographically coupled with the session-key at the time of key generation in the KDC. For this, the generated hash function, master key and the session-key are used for producing the encrypted session-key, which has to be transferred. All the operations have been performed using the C programming language. This process can be widely applicable to all sorts of electronic transactions online or offline; commercially and academically.

Key words: Encryption, decryption, master key, session-key, key distribution center and control vector

INTRODUCTION

In network transactions, encrypted communication between the end systems using a temporary key known as session-key. The session-key is used for the duration of a logical connection, such as frame relay connection, or transport connection. Each session-key is obtained from the KDC over the same network facilities used for end-user communication. For two parties, say A and B, key distribution can be done in a number of ways: (a) a session-key can be selected by A and physically delivered it to B, (b) a third party can select the session-key and physically deliver it to A and to B, (c) if A and B have previously and recently used a session-key, one party can transmit the new session-key to the other, encrypted it using the old session-key and (d) if A and B each has an encrypted connection to a third party C, C can deliver a key on the encrypted links to A and to B. The first two options call for a manual delivery of a session-key. In the third option, if an attacker ever succeeds to gaining access to one session-key, then the entire subsequent session-key will be revealed. Furthermore, the initial distribution of potential millions of session-keys must still be made. The fourth option is now widely used where the third party is known as the Key Distribution Center (KDC). The concept of a key hierarchy and the use of automated key distribution techniques greatly reduce the number of keys that must be managed and distributed. KDC performs the key distribution process by using the control vector encryption and the control vector decryption techniques. The conventional technique for control vector encryption is described in section 3.1 and the proposed improved technique is discussed in section 3.2. Here, Control Vector (CV), master key (K_m), Hash function (H) and the session-key K_s are used as key parameters.

The control vector is a random binary bit string. Each session-key has an associated control vector consisting of number of fields that specify the uses and restrictions for that session-key. The length of control vector may vary. The control vector is cryptographically coupled with the key at the time of key generation in the KDC. When a session is delivered to a user from the KDC, it is accompanied by the control vector (CV) in clear form. The master key (K_m) is like the control vector with a difference is that its length is fixed. The encrypted session-key can

Corresponding Author: M. Ismail Jabiullah, Department of Computer Science and Engineering, Research Student, Dhaka University and Associate Professor, Institute of Science and Technology, Dhaka 1209, Bangladesh

be recovered only by using both the master key and the control vector. The CV and the K_m is generated by the KDC for each session-key. The sender and the receiver must know them through the KDC. KDC provides the CV and the K_m to the sender and produces an Encrypted Session-key (Eks) using the encryption method DES. The receiver also takes the CV and the K_m from the KDC and the Encrypted Session-Key (EKs); sent by the sender and generate the actual session-key (K_s) through the decryption method.

MASTER KEY GENERATION

For any given plaintext message, the fixed length master key is generated using the pseudorandom bit stream generation technique. The different random bit stream is produced as constant output and then the output is used to generate random bit stream and finally generated the fixed length pseudorandom bit stream that is used as the master key. Here the length of the generated master key is 128.

KDC FOR SESSION-KEY DISTRIBUTION

A key distribution center is responsible for distributing keys among the pairs of users as needed. Each user must share a unique key with the key distribution center for purposes of key distribution (Fig. 1).

At least two levels of keys must be used:

- communication between the end systems is encrypted using a temporary key, often referred to as a session-key and
- session-key is transmitted in encrypted form, using a master key that is shared by the KDC and an end system or user (Fig. 2).

Each user must share a unique master key with the KDC. If there are N end users, N(N -1)/2 session-keys are needed at any one time, but only N master keys are required. The master key can be distributed in a non-cryptographic way, such as physical delivery. Now, for large networks, a single KDC may not be adequate. A hierarchy of KDCs can be established where each local KDC is responsible for a small domain of the overall network. If the two parties of an exchange are within the same local domain, their local KDC is responsible for key distribution. Otherwise, the corresponding local KDCs can communicate through a global KDC. Any one of the three KDCs involved can select the key.

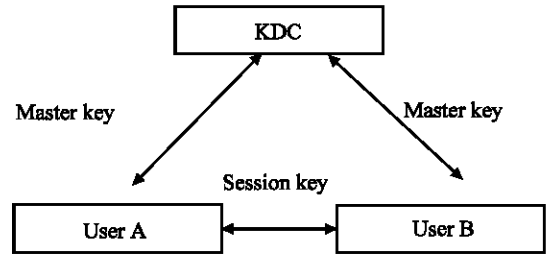


Fig. 1: Key distribution

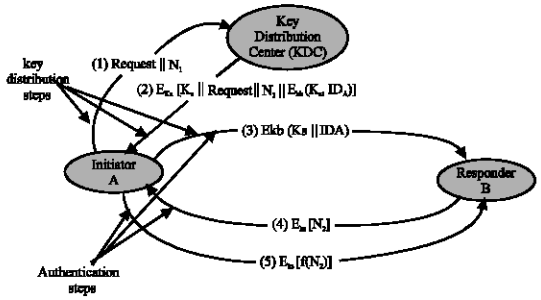


Fig. 2: Key distribution scenario using KDC

Conventional techniques: The control vector is inputted to the hash function and the hash output and the used K_m are XORed and the resultant is encrypted with a technique using K_s. The encrypted K_s is transmitted to the destination. In the receiving end, the control vector is inputted to the hash function; and the generated hash code and the master key are XORed and the resultant and the received encrypted session-key EK_s is inputted to the decryption function. The output of the decryption function is the transmitted session-key that is to be distributed. The conventional technique is depicted in the Fig. 3.The main drawback of the technique is that one can easily analyze the key, K without knowing both the CV and K_m by hitting the encryption function. This drawback can be overcome by the modified technique, where the key is hiding from the intruders by increasing the level of encryption. Thus it became more secure and reliable.

The K_s is to be distributed through the KDC using the control vector encryption/decryption process (Fig. 4). The control vector CV is considered as a pseudorandom bit stream whose length is not fixed. The K_m is considered as a fixed length pseudorandom bit stream whose length is 128. The control vector is passed through hash function MD5 that produces a hash value h whose length is also 128, equal to the length of the K_m. The control vector is inputted to the hash function MD5. The hash output and the K_s is then used in the encryption

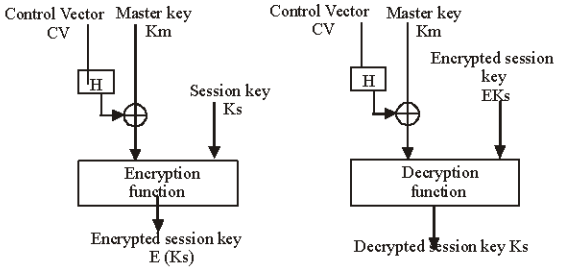


Fig. 3: Conventional control vector encryption/ decryption process

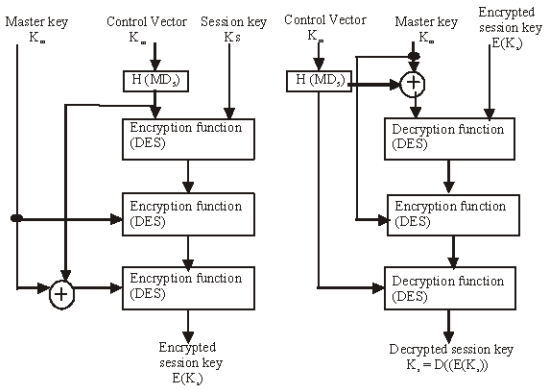


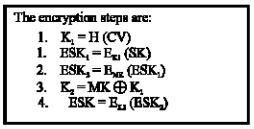
Fig. 4: Control vector encryption and decryption process

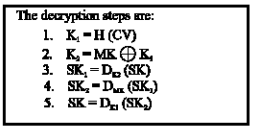
technique DES. The encrypted output is again encrypted using the same K_s. Then the XORed output of the K_s and the hash output, and the previously encrypted output are again inputted to the encryption function DES. The resultant of the encryption function is then transmitted to the destination. In the receiving end, the control vector is inputted to the hash function MD5 and the hash output and the K_m is XORed and inputted to the decryption function DES with the received encrypted EK_s. The decrypted output is again decrypted using the DES with the K_m. Finally, the distributed K_s is found by decrypting the previous calculating result with the hash output.

IMPLEMENTATION

To implement the improved algorithm, CV is used as a pseudorandom bit stream, whose length is 128. MD5 is used as the hash function and DES is used as the encryption decryption function. The encryption algorithm and the decryption algorithms are given below:

Encryption algorithm: Here CV = control vector, H = MD5, hash output, Ks = session-key and the K_m = master key.





Input/Output analysis: If the plaintext message This is the sample message for encryption and decryption. is given as the session-key, then the produced XORed output is

00110110110001000001100111110001100110011100101001
10010010100011101111101001111000110001001101101110
00000101101110101010100101101101011101101100111001
10010111011111011110111111111111011001010111111111
11101011100010111111111101111011110001101100110101
11010" and finally the produced encrypted session-key is

"ÜÞÞàÚÚÞÝÝÛÚÝÞÚàÛÞÞÛÝÛÝÝÞÝÛÞÜÜÛÜÜÜßÞß
ÙÙÝÝÜÜÚÝÞÙàÜßÞÛÜÜÝÞÝÞÛÝÜÜÛÜÜ".

In the receiving end, the session-key is found by performing the decryption algorithm in reverse order.

Complexity analysis: The complexity of this technique lies between the level of encryption and the length of the keys. Here, the key length is used as 128 and the level of encryption is used as 3. So, the complexity of the key space for this technique is calculated as $2^{128} \times 2^{128} \times 2^{128} = 2^{384}$. The complexity of each level:

Encryption Level	Complexity
Level 1	3.4028236692093846346337460743177e+38
Level 2	1.1579208923731619542357098500869e+77
Level 3	3.9402006196394479212279040100144e+115

CONCLUSION

Session-key distribution using a third party key Distribution Center (KDC) will give the end users an enormous strength of secrecy. The strategies used in this encryption/decryption process are mainly in two schemes. One is tag scheme and the other is control vector scheme. The tag scheme is used in earlier days. It causes some drawbacks. These drawbacks have been overcome by using the control vector scheme. The control vector schemes gives the end-users, freedom on the length and the clarity on the form, in which it may appear. The encryption function used in this improved scheme doesn't need to be highly complex. Because, the session-key has been transmitted here and the lifetime of

the session-key is too small to be unsecured. The scheme has been implemented using C language. This scheme has been used as a teaching tool for a practical adaptation. While key notarization may be viewed as a mechanism for establishing authenticated keys, control vectors provide a method for controlling the use of keys, by combining the idea of key tags with the mechanism of simple key notarization. Cryptographically binding the control vector C to S at the time of key generation prevents unauthorized manipulation of C, assuming only authorized parties have access to the key-encrypting key K.

REFERENCES

1. Menezes, A., P. Van Oorschot and S. Vanstone, 1997. Handbook of Applied Cryptography, CRC Press, Inc.
2. Ellis Horowitz, Sartaj Sahni and Sanguthevar, Computer Algorithms, Galgotia Publications pvt. Ltd.
3. Kenneth, H. Rosen, 1984. Elementary Number Theory and its Applications, Addison-Wesley,
4. Seymour Lipschutz and Marc Lars Lipson. Theory and Problems of Discrete Mathematics, 2nd Edition, Schaum's Outline Series, McGraw-Hill.
5. Goldwasser, S. and B. Mihir. Lecture Notes on Cryptography, MIT laboratory of Computer Science, 545 Technology Square, Cambridge, MA 02139, USA.
6. William Stallings, Cryptography and Network Security-Principles and Practice, 2nd Edn., Prentice Hall Upper Saddle River, New Jersey 07458, ISBN: 981-403-589-0.
7. William Stallings, Data and Computer Communications, 6th Edn., Prentice Hall International Inc., ISBN: 0-13-086-388-2.
8. Lenstra, H.W., 1987. Factoring Integers with Elliptic Curve-Annuals of Mathematics, 126: 649-673.
9. Menezes, A., 1993. Elliptic Curve Public Key Cryptosystems, Kluwer Academic Publishers.
10. Andrew, S. Tanenbaum, 2000. Computer Networks, Third Edition, Prentice Hall of India Private Limited, New Delhi-110 0001.
11. Yeuan-Kuen Lee, Confidentiality Using Conventional Encryption.
12. Ismail Jabiullah, M., Sk. Mizanur Rahman, M. Lutfar Rahman and M. Alamgir Hossain, 2004. Pseudo-random Bit String for Cryptographic Applications, The Dhaka University J. Sci.