ISSN 2286-4822
www.euacademic.org

Impact Factor: 3.1 (UIF)
DRJI Value: 5.9 (B+)

Propose Key Generation Framework for Block Cipher Algorithm

SAMEERAH FARIS KLEBUS
Computer Science Department, College of Science
Karbala University, Karbala
Iraq

Abstract:

Good cryptographic systems should always be designed so that they are as difficult to break as possible. This research will concentrate on increasing the complexity of block cipher encryption. This complexity could be done by building a dynamic pool that consists of thousands of fused random bits; these bits come from previously used key and many other resources.

Two fusion methods are proposed to generate the block cipher keys. The first method is by using Artificial Neural Network (ANN) and the second one by using genetic Algorithm (GA). For these two methods we introduce an example for Data Encryption Standard (DES) algorithm.

Hamming distance function is used as threshold for testing the reliability of the keys in the proposed methods. This research also proposed a method for generating the sub-keys for the Blowfish algorithm.

Key words: block cipher encryption, key generation framework, Artificial Neural Network (ANN), Genetic Algorithm (GA)

General Background:

Thanks to the Internet make the world as a global village and all the doors are unlocked. This openness-nature of the Internet comes from the infrastructure and standard protocols (Transmission Control Protocol /Internet Protocol suite). Internet sites like government and commercial agencies suffer from impact of security breaches. Their secure, confidential, and sensitive information are vulnerable to any person connected to the Internet. Unauthorized persons called Hackers and Crackers may be intend to stole or destroy the secure information of these agencies. This would cause real damage for the attacked agency.

For that it is necessary to protect information from different threats and the intervention of the unauthorized users. The one solution is to build secure system which tries to fulfilling all the requirements of the security and must protect the data not only inside the Internet site for an agency, but also protect data which it transmitted from and to the protected Internet site. Cryptography: Is the science and study of secret writing. Crypt analysis: Is the science and study of methods of breaking encrypted messages. Encryption (Enciphering): Is the operation of disguise confidential information in such a way that its meaning is unintelligible to unauthorized. Decryption (Deciphering): Is reverse operation of encryption by transform unintelligible information to the original confidential information. The basic elements for encryption are key and algorithm. According key used we can classify the encryption to two types Private Key Encryption and Public Key Encryption. Private Key Encryption: Also called symmetric algorithms, the encryption and decryption keys are identical. So, key must be secure “private”. Data Encryption Standard (DES) is the most famous encryption method in this type. Its private key encryption algorithm has 64-bit key, encrypts data 64-bit at a time. And decrypts data 64-bit-key at a time. Public Key Encryption: Also called A symmetric algorithm, the encryption and decryption keys are differs. So key must not be secure because, there are two keys. Each person has a private key, which he uses. To decrypt or digitally sign messages and a public key which others use to send messages to him, or to verify his digital signature. According algorithm used we can classify the encryption to two types Stream Cipher Algorithm and Block Cipher Algorithm. Stream Cipher Algorithm: Stream cipher is one of the simplest methods of data encryption. When a stream cipher is employed. Each bit of the data is sequentially encrypted using one bit of the key. A classical example of a stream cipher was the Vernam cipher used to encrypt Teletype traffic. Block Cipher Algorithm: Block cipher unlike stream cipher, which encrypts every single bit, block ciphers are designed to encrypt data in chunks of specific size. A block cipher specification will identify how much data should be encrypted each pass (called a block) as well as what size should be applied to each block. DES algorithm and Public key encryption are the famous examples of this type.

Proposed Key Generator

There are many traditional methods to generate random numbers for use in cryptographic keys. The point is that the data must be unpredictable for any external observer. The most famous methods as widely known are:

Conventional random number generators available in most programming environments but are not suitable for use in cryptographic applications.

Obtain some environmental noise from device latencies, resource utilization statistics, network statistics, or keyboard interrupts.

Cryptographic pseudorandom generators typically have a large pool ("seed value") containing randomness. Bits are returned from this pool by taking data from the pool, optionally running the data through a cryptographic hash function to avoid revealing the contents of the pool. When more bits are needed, the pool is stirred by encrypting its contents by a suitable cipher with a random key (that may be taken from an unreturned part of the pool) in a mode which makes every bit of the pool depend on every other bit of the pool.

New environmental noise should be mixed into the pool before stirring to make predicting of previous or future values even more impossible.

The proposed method for generating keys will be explained in the following steps:

Step one:

Since the proposed method aims to generate strong keys for Block Cipher algorithms so we must avoid weak keys (e.g. in DES keys of all 1s or 0s, encrypting twice decrypts). With many block ciphers there are some keys that should be avoided, because of reduced cipher complexity; these keys are such that the same as the generated sub-key in more than one round. Weak Keys are the same sub-key is generated for every round. DES has 4 weak keys. Semi-Weak Keys only two sub-keys are generated on alternate rounds, DES has 12 of these (in 6 pairs). Demi-Semi Weak Keys have four sub-keys generated. None of these cause a problem since they are a tiny fraction of all available keys. However they must be avoided by any key generation program.

Step two

Build a dynamic pool that contains a huge number of fused used keys. These keys fused with many generated streams of random bits. This pool will be updated in two cases:

The pool must be updated when we take a key from it, so we must add random bit to be fused with bits founded in the pool.

The pool must be updated when we add a used key to be fused in the pool.

The proposed pool-based-fusion technique used with the dynamic pool could be summarized in the following steps:

Dynamic pool in the proposed method is a binary file.

Initially put a huge number of random bits from many resources in the pool.

The fusion will depend on permutation process that will be updated from time to another.

Step Three:

To generate keys for Block Cipher encryption algorithm the proposed method introduces three proposals all of them depending on dynamic pool for picking up the seed of random generated keys, these proposals are:

First proposal:

This proposal aims to generate a random key for any block cipher algorithm, here we will explain the proposal with DES, that by using a proposed artificial neural network as the following:

Take a seed (seed has number of bits equal to 64 bit) from dynamic pool.

Since the no. of bits needed for the generated DES keys must be 64 bits, we suggest the following architecture for the proposed ANN, see figure (1).

No. of inputs nodes will be 64 nodes.

No. of outputs nodes will be 64 nodes.

No need for hidden layers.

Matrix of 64*64 for initial weights

For processing function the binary sigmoid function will be used, which presented by any equations.

For accepting the output pattern (64 bit) as a good key for DES that will be submitted to hamming distance function to compare it with known weak keys. The hamming distance must have biggest values to make the output pattern accepted as a DES key. Hamming distance is the number of bit positions in which two bit patterns differ. Starting with a complete list of legal keys, we need to find the keys whose Hamming distance is the biggest.

Second proposal:

The proposal aims to generate a random key for any block cipher algorithm, here the proposal with DES will be explained, that by using a Genetic algorithm as the following:

Take 200 seeds (each seed has number of bits equal to 64 bit) from dynamic pool.

An evaluation function that plays the role of the problem environment (best key), rating solutions in terms of their "fitness". Here the proposed evaluation function for each key is the hamming distance function that to compare the keys with known weak keys.

Genetic operators that alter the composition of offspring. One-point crossover is the most suitable crossover operator, where a crossover points on the genetic code are selected randomly, and two parent frames are interchanged at these points.

Crossover exploits existing keys potentials, but if the population does not contain all the encoded information needed to find the best key, no amount of keys mixing can produce a satisfactory solution. For this reason, a mutation operator capable of spontaneously generating new key is included. The most common way of implementing mutation is to flip a bit with a probability equal to a very low, given Mutation Rate (MR). A mutation operator can prevent any single bit from converging to a value through the entire population and, more important, it can prevent the population from converging and stagnating at any local optima.

Values for the various parameters that the genetic algorithm uses population size, rate of applied operators, etc. In the particular problem the following parameters of the genetic algorithm are used: Population size, pop-size = 200 (the parameter was already used), Probability of crossover, PC = 1, Probability of mutation, PM = 0.001 (the parameter will be used in a mutation operation).

Continue with genetic processing until the optimized key to be the master key will be obtained.

Third Proposal:

Since blowfish uses a large number of subkeys. Traditionally these subkeys must be precomputed before any data encryption or decryption.

The P-array consists of 18 of 32-bit subkeys: P1, P2,..., P18.
There are four 32-bit S-boxes with 256 entries each:
S1,0, S1,1,..., S1,255;
S2,0, S2,1,..,, S2,255;
S3,0, S3,1,..., S3,255;
S4,0, S4,1,..,, S4,255.

Generating the Subkeys:
This proposal aims to generate a random key for blowfish by modifying the method of generating the sub-keys, as the following:

From dynamic pool initialize first the P-array and then the four S-boxes, in order, with a fixed string. This string consists of the hexadecimal digits of pi . For example:

P1 = 0x243f6a88
P2 = 0x85a308d3
P3 = 0x13198a2e
P4 = 0x03707344………

XORing P1 with random selected 32 bits from the pool as a first 32 bit of the key. XORing P2 with random selected 32-bits from the pool as a second 32 bit of the key, and so on for all random selected 32 bits of the key (possibly up to P14). Repeatedly cycle through the key bits until the entire P-array has been XORed with all random selected 32 key bits.

Encrypt the random selected string from dynamic pool with the Blowfish algorithm, using the subkeys described in steps (1) and (2).

Replace P1 and P2 with the output of step (3).

Encrypt the output of step (3) using the Blowfish algorithm with the modified subkeys.

Replace P3 and P4 with the output of step (5).

Continue the process, replacing all entries of the P-array, and then all four S-boxes in order, with the output of the continuously-changing Blowfish algorithm.

In total, 521 iterations are required to generate all required subkeys. Applications can store the subkeys rather than execute this derivation process multiple times.

Conclusions

Building dynamic pool based fusion to enable the administrators of secure system from reusing the previously used keys.

Building system for generating secure keys for block cipher algorithms make the encryption system much more reliable since it provides powerful keys with very little time.

Using proposed ANN and proposed GA aid permutation and randomize the bits of keys. Also efficiently testing the introduced keys with the weak keys using hamming distance function.

Modifying the sub-keys generation of blowfish algorithm by taking all the bits of initial key from the dynamic pool, that making the key more strong than take seed A of the initial key then complete it by duplicating it AAAA and so on.

REFERENCES

Aiazzi, B., L. Alparone, S. Baronti, and A. Garzelli. 2002. “Context-driven fusion of high spatial and spectral resolution data based on oversampled multiresolution analysis.” *IEEE Trans. Geosci. Remote Sensing* 40(10): 2300–2312.

Al-Hamami, A.H, Mohammad, A.Al-Hamami, & Soukaena, H. Hashem. 2006. "A Proposed Modifications to Improve the Performance of Blowfish Cryptography Algorithm." *First National Information Technology Symposium (NITS 2006) Bridging the Digital Divide: Challenge and Solutions*. King Saud University, Riyadh, Kingdom of Saudi Arabia, 5-7 Feb.

Beckett, B. 1997. “Introduction to cryptography and PC security.” McGraw-Hill companies.

Chatzigiannakis, V., S. Papavassiliou, G. Androulidakis, B. Maglaris. 2006. “On the realization of a generalized data fusion and network anomaly detection framework.” *Fifth International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP’06)*, Patra, Greece, July 2006.

Laporterie-D´ejean, F., H. de Boissezon, G. Flouzat, and M.-J. Lef`evre-Fonollosa. 2005. “Thematic and statistical evaluations of five panchromatic/multispectral fusion methods on simulated PLEIADES-HR images.” *Inform. Fusion* 6(3): 193–212, Sep. 2005.

Mohammad, A. Al-hamami & Soukaena H. Hashem. 2006. “Improving performance and random signature schemes in twofish cryptosystem.” *Journal of Al_Rafidian.*

Stalling, W. 2001. “Network security essential: application and standard.” William stalling books for network and data communication technology.

Wang, Z., D. Ziou, C. Armenakis, D. Li, and Q. Li. 2005. “A comparative analysis of image fusion methods.” *IEEE Trans. Geosci. Remote Sensing* 43(6): 1391–1402, June 2005.