NEW SYMMETRIC ENCRYPTION SYSTEM BASED ON EVOLUTIONARY ALGORITHM

A. Mouloudi

Department of Computer Sciences, Ibn Tofail University, Kenitra, Morocco

ABSTRACT

In this article, we present a new symmetric encryption system which is a combination of our ciphering evolutionary system SEC [1] and a new ciphering method called "fragmentation". This latter allows the alteration of the appearance frequencies of characters from a given text. Our system has at its disposed two keys, the first one is generated by the evolutionary algorithm, the second one is generated after "fragmentation" part. Both of them are symmetric, session keys and strengthening the security of our system.

KEYWORDS

Cryptography, cipher, symmetric encryption, evolutionary algorithm.

1. INTRODUCTION

Basic cryptographic algorithms split into two families: symmetric algorithms, otherwise known as secret-key algorithms, which normally require a key to be shared and simultaneously kept secret within a restricted group, and public-key algorithms where the private key is almost never shared. From outside, this may give the impression that symmetric techniques become obsolete after the invention of public-key cryptography in the mid 1970's. However, symmetric techniques are still widely used because they are the only ones that can achieve high-speed or low-cost encryption. Today, we find symmetric algorithms in GSM mobile phones, in credit cards, in WLAN connections... Therefore, symmetric cryptology is a very active research area which is stimulated by a pressing industrial demand for low-cost implementations (in terms of power consumption, gate count...).

In the article [1], we presented a new symmetric encryption system called ciphering evolutionary system (SEC). This system is based on techniques of the evolutionary algorithm [2],[3],[4]. This article was followed by another article, where we introduced a new system based on SEC [5], called "fusion" which is safer.

In the article [6] we have generalized the SEC system at any chain formed of blocks of bits.
In this paper we present a new encryption system based on the SEC system. This new system uses a different technique called "fragmentation". It proceeds by a change of frequency of occurrence of characters of the clear message, using the SEC algorithm. After, it uses the fragmentation technique in order to have the character positions lists with approximately the same size, by a fragmentation of lists which have larger sizes, adding new characters.

2. SYSTEM DESCRIPTION

Let M the message to be encrypted, consisting of n characters.

2.1. Problem formalization

c_1, c_2, ..., c_m are the different characters of M ($m \leq 256$). Denote by L_i ($1 \leq i \leq m$) the list of positions of the character c_i in M and Card(L_i) the number of his occurrences in M.

Note: $L_i \cap L_j$ is empty, for $i, j \in [1, m]$, with $i \neq j$.

L_1, L_2, ..., L_m is a partition of the set {1, 2, ..., n}.

The message M can be represented by the vector below:

Table 1: Representation of the message by a vector

(c_1, L_1)	(c_2, L_2)	…	(c_m, L_m)

Our goal is to change the maximum frequency of appearance of characters in the message M and thus establish the more disorder in their positions. For this, we change iteratively distribution lists on the different characters of M such that the difference between the cardinal of the new list L'i assigned to character ci and the cardinal of the initial list Li of ci is maximized. There, we realize that we are in front of an optimization problem. As evolutionary algorithms have proven effective in solving this problem, so we are appealing to these algorithms, including those applied to permutations problems.

2.2. First encryption: Application of the SEC algorithm

Step 0: Coding
An individual (or chromosome) is a vector of size m. Genes are the L_{pi} lists ($1 \leq i \leq m$). The L_{pi} th gene contains the new positions will be taken by the character.

Step 1: Creating the initial population P_0 composed of q individuals: X_1, X_2, ..., X_q.

We call initial Ch-chromosome genes the lists L_1, L_2, ..., L_m (placed in that order) that represent the message before the application of the algorithm.
We apply q permutations on Ch_initial in order to obtain an initial population who is q potential solutions to the problem.
i: = 0.

Step 2: Evaluation of individuals

X_j is a P_i individual genes are: L_{j1}, L_{j2}, ..., L_{jm}.
We define the evaluation function F of all individuals by $F(X_j)$:

$$F(X_j) = \sum_{i=1}^{m} \left| \ \text{card}(L_{j_i}) - \text{card}(L_i) \ \right|$$

Step 3: Selection of the best individuals

We use the traditional method of the wheel to retain the strongest individuals.
We introduce a control function that will eliminate individuals for which only minority of genes have changed values compared to the Ch_initial original chromosome.
Since we are brought back to a problem of permutations with constraints, we then apply genetic operators adapted to this kind of problem.

Step 4: Applications of genetic operators

- Crossing MPX (Maximal Preservative X)

This crossing is applied to selected individuals with a specific rate. According to [7] the best rate is on the order of 60% to 100%.

- Transposition mutation:

We choose the mutation that is to randomly switch between two genes on a chromosome. This operator is applied to the individuals produced by crossing with a suitable rate, preferably from 0.1% to 5% [7], [8]

Place new offspring in a new population P_{i+1}.
Repeating steps 2, 3 and 4 until a stop criterion.

Set the stop condition:
The function F is bounded for any individual X_k. Indeed,

$$F(X_k) = \sum_{i=1}^{m} \left| \ \mathrm{card}(L_{k_i}) - \mathrm{card}(L_i) \ \right|$$

$$\leq \sum_{i=1}^{m} (\mathrm{card}(L_{k_i}) + \mathrm{card}(L_i)) \leq \ 2 * n$$

Theoretically and mathematically speaking, the function F admits a maximum since it is bounded. According to some research findings convergence of an evaluation function is provided. This is confirmed in the working examples.

Last phase of our algorithm:

Final_Ch denote the final solution given by our evolutionary algorithm. Swapping that can go from Ch_initial to Ch_final is our symmetric key. This key will be called genetic key.

2.3. Second encryption: Fragmentation of the lists

2.3.1 Informal description of the algorithm

The message M' obtained in the first part consists of the same characters c1, c2, ..., cm in M which are associated lists L'1, L'2, ..., L'm. These lists are generally of different sizes. The purpose of the second part of our system is to transform the message M' in a message M" whose lists associated with its characters are almost the same size, this by breaking down the lists of larger sizes in other lists, which are associated with new characters.

2.3.2 Formal Description

Table 2: The fragmentation of the lists

$L'_1, L'_2, ..., L'_m$ are the lists of different positions associated with the characters $c_1, c_2, ..., c_m$ in the message M', obtained from the first encryption. Let

$$l = E\left[\frac{\sum_{i=1}^{m} card(L'_i)}{m}\right]$$

Among the lists above, we can distinguish those large sizes (their sizes are above average size l).

If L'_j is a large list associated with the character c_j (card $(L'_j) > l$)

Then let $n_j = E\left[\frac{card(L'_j)}{l}\right] + 1$

We will break the list L'_j into n_j lists $L'_{j_1}, L'_{j_2}, ..., L'_{j_{n_j}}$ with equal size:

The first list L'_{j_1} will be associated with the character c_j while the other lists are associated with new characters $c_{j_2}, c_{j_3}, ..., c_{j_{n_j}}$ therefore $L'_j = L'_{j_1} \cup L'_{j_2} \cup ... \cup L'_{j_{n_j}}$

The distribution of L'_j on $L'_{j_1}, L'_{j_2}, ..., L'_{j_{n_j}}$ is carried on as follows:

If $L'_j = \{p_1, p_2, ..., p_{m_j}\}$ Then

$L_{j_1} = \{p_1, p_{n_j+1}, p_{2n_j+1}, ...\}$

$L_{j_2} = \{p_2, p_{n_j+2}, p_{2n_j+2}, ...\}$

…… ……

$L_{jn_j} = \{p_{n_j}, p_{2n_j}, p_{3n_j}, ...\}$

End If

End If

The character associated with the list who's underwent fragmentation is saved with the new characters in a list called fragment key.

The above list will be a secret key that will be used during decryption.

In a formal way, the fragmentation cipher algorithm is:

Table 3: The fragmentation cipher algorithm

Data: Message M', obtained from the evolutionary algorithm SEC

Results: Encrypted Message M" and the fragment key

Beginning of the algorithm

Determine the lists of character positions $L'_1, L'_2, ..., L'_m$ associated with characters $c_1, c_2, ..., c_m$ of the message M'.

Let $l = E\left[\frac{\sum_{i=1}^{m} card(L'i)}{m}\right]$, where E [x] is the integer part of x (ie the largest integer less than or equal to x).

For j: = 1 to m do

If $card(L'_j) > l$ then / * L'_j is a large list * /

Let $n_j = E\left[\frac{card(L'_j)}{l}\right] + 1$ and $m_j = card(L'_j)$

/ * Fragmentation of L'_j */

Let n_j lists $L'_{j_1}, L'_{j_2}, ..., L'_{j_{n_j}}$ initially empty, such as L'_{j_1} is associated with c_j and $L'_{j_2}, ..., L'_{j_{n_j}}$ are associated respectively with novels characters $c_{j_2}, ..., c_{j_{n_j}}$

/ * Distribution of L'_j on $L'_{j_1}, L'_{j_2}, ..., L'_{j_{n_j}}$ */

Suppose $L'_j = \{p_1, p_2, ..., p_{m_j}\}$

For k = 1 to m_j do

$d := k \bmod(n_j)$ (i.e, d is the remainder of the integer division n_j by k) Move p_k to list L'_{j_d}

End of For

Add $(c_j, \{c_{j_2}, ..., c_{j_{n_j}}\})$ to fragment key

End If End of For End of the algorithm

2.4 Decryption

Decryption is performed in two essential steps:

First step: From the message M" and using fragment key we find the message M'.
Second Step: From the message M' and using genetic key we find the message M.

2.4.1 First decryption

For each key-element fragment $(c_j, \{c_{j_2},...,c_{j_{n_j}}\})$ we replace in M" all occurrences of the characters $c_{j_2}, c_{j_3},...,c_{j_{n_j}}$ by character c_j. This gives the message M'.

2.4.2 Second decryption

Decrypting the message M' is done in two steps:

In the first step, we represent the cipher text, by a vector of lists as we have done for the plaintext message. Denote c'_1, c'_2, ..., c'_m the different characters of M' and L'_1, L'_2, ..., L'_m the lists their respective positions in M'. In the second step, through genetic key, the characters will find their corresponding position lists in the plaintext message. Indeed, the genetic Key is a permutation of {1, 2, ..., m} that we can represent by a vector with size m, denoted Key, such that: Key (1) = p_1, Key (2) = p_2, ..., Clef (i) = p_i, ..., Clef (m) = p_m where The c'_{p1} character will be associated with the list L'_1, the c'_{p2} character will be associated with the list L'_2.

The c'_{pm} character will be associated to the list L_m. Thus, we get the message M.

3. EXPERIMENTATION

We applied our algorithm to several messages of different sizes, and for each of them, we make the application for different sizes of population. Then we record the value of the convergence of the fitness function, the number of the generations reached at the time of this convergence. Thus we can select the size of population that gives best value of evaluation function.

In this paper, we consider for example two messages. For each message we applied, first, the SEC algorithm, leading to a new message M ', after using the deduced genetic key. Then, and as the second step of encryption, we cut the lists of positions associated with the most frequent characters of the message M into lists hose size is in the average. These new lists will be assigned to new characters, to finally obtain the encrypted message M ".
First message: (The Size is 882)

Table 4: The first message

In cryptography, encryption is the process of encoding messages or information in such a way that only authorized parties can read it. Encryption does not of itself prevent interception, but denies the message content to the interceptor. In an encryption scheme, the intended communication information or message, referred to as plaintext, is encrypted using an encryption algorithm, generating ciphertext that can only be read if decrypted. For technical reasons, an encryption scheme usually uses a pseudo-random encryption key generated by an algorithm. It is in principle possible to decrypt the message without possessing the key, but, for a well-designed encryption scheme, large computational resources and skill are required. An authorized recipient can easily decrypt the message with the key provided by the originator to recipients, but not to unauthorized interceptors.

After applying of the first step of the encryption algorithm we obtained the genetic key.

Table 5: Genetic key of the first message

14 15 16 17 11 12 4 5 6 7 8 25 26 27 28 29 30 31 32 0 1 2 3 9 10 18 13 21 19 20 22 23 24

Using this key the following message is obtained:

Table 6: Message deduced, after first step of encryption

Iuwlzytoggrpehy, lbcryttgon iI the pooaesnnla ebcoding mvbnages or lnf amvE goawln luFz a w-E thatglnly luFgrapzed larties czE rpbdclt. Enwzypogon doeb lwa of ltself loevhbw interceptgoaw buF deniIbnlhe mesnnEe lzntgbw to thebln webcebtor. In aE lbwzyptgIn scheme,ylge iIwgbded lzmmuniczEgoa inf amaEg on orpmebsnEe, rpbebred tg aE llxEIwexegyls lbwzypted uFnng aE enczyttgon wllgoriIgm, gebwbating lzphebtexe thrE canwlaly be rpbE if debzytoed.qloa tec hniIal rpbEonw, aE encryptgInwlnzeme uFuFlly usnb a loeudo-spEwomvencryp ogIn key g.berated by lEwalgoapthm. Iu lI in loiIwzpleblosnnblxbtg decrypt lge mesnnE.bwithout losnesnnng lge key, but,yl a a w-ll-designed ebwzyption lchrb e,ylxEge lzmvuFatgInwl lesourczb aEw skillxlEe lequired. An authorpzkb lebzp ienwgczE eaEnly dcbzytt the mesnnEebwiIg thrbkey prpvided bytlhe oapgin aEor tg rpbiIoents, buFglwt to unwuFgrapzed interpzbtorp..

After applying second encryption step we obtain the following fragmentation key:

Table 7: Fragmentation key deduced

(i ; (▼,!)); (s; (")); (r; (#, $)); (p; (%)); (t; (&,')); (o; ((,)); (n; (*,+)); (; (/,0,1,2)); (c; (3)); (a; (4)); (e ; (5,6,7))

And the following encrypted message:

Table 8: Message deduced after the second step of encryption

Iuwlzytogg#pEhy,/lb3$yt&g(+0I1'h52%o)a6"nnla/7b3(d!*g0mvbn4g5"1)#2l+f amvEg
(awl*/luFz041wE2&h4'gl+ly/luFgrapz6d0l4#&7"13zE2$pbdcl'./E*wzy%og(+0d)5b1lwa2(
f l'"6lf/lo7vhbW0!+&5$36%'g)aw1buF2d7*Ibnlh5/m6"nnE70lz+'gbw1&(2'h5bl*w6b37b&
)#./I+04E1lbwzy%'gI+2"3h5m6,ylg7/Iwgbd5d0lzmmu+!3zEg)a1*f am4Eg(+2)#pm6b"nE7
,/$pb5b#6d0&g14E2llxEIw7xegyl"/lbwzy%&5d0uFn+g14E26*3zyt'g(+wllg)#Igm,/g7bw
b4&!*g0lz%h5b'6xe1&hrE234+wlaly/b70#pbE1f2d5bzyto6d.ql(a/&73h*!I4l0#pbE)
+w,14E25*3$y%'gI+wlnz6m7/uFuFlly0u"nb142lo5ud(spew)mv6*3#y%ogI+/k7y0g.
b5$4&6d1by2lEw4lg(ap'hm./Iu0lI1+2lo!Iwz%l7bl)"nnblxb&g0d53$y%'1lg62m7"
nnE.bw!&h(u'/l)"n5"nn*g0lg61k7y,2bu&,yla/40wlld5"g+6d17bwzy%'!(*2l3hrb5,ylxEg6
/lzmvuF4&gI+wl0l7"(u#3zb14Ew2"kllxlE5/l6qu!#7d.0A*14u'h)$pzkb2l5bz%!6+wg3zE
/74Enly0dcbzyt&1'h52m6"nnE7bwIg/&hrbk5y1%$pv!d6d2bytlh7/(apg4E)#0&g1$pb
!Io5+'",2buFglw&/')0u*wuFgrapz6d1!+&7#pzb'($p..

The second message: (The size is 1028)

Table 9: The second message

Encryption, by itself, can protect the confidentiality of messages, but other techniques are still needed to protect the integrity and authenticity of a message; for example, verification of a message authentication code (MAC) or a digital signature. Standards for cryptographic software and hardware to perform encryption are widely available, but successfully using encryption to ensure security may be a challenging problem. A single error in system design or execution can allow successful attacks. Sometimes an adversary can obtain unencrypted information without directly undoing the encryption. See, e.g., traffic analysis, TEMPEST, or Trojan horse.

Digital signature and encryption must be applied to the ciphertext when it is created (typically on the same device used to compose the message) to avoid tampering; otherwise any node between the sender and the encryption agent could potentially tamper with it. Encrypting at the time of creation is only secure if the encryption device itself has not been tampered with.

After applying of the first step of the encryption algorithm we obtained the genetic key:

Table 10: The genetic key of the second message

10 11 12 13 6 7 8 9 14 38 0 1 2 24 25 26 27 28 29 30 31 32 33 34 35 36 37 3 4 5 15 16 17 18 19 20 23 21 22

Using this key the following message is obtained:

Table 11: Message deduced, after first step of encryption

E bsMpCypn,iby itycrf, cs. proters Ehe cotfipwntyalityAEf mescagTr, but other terhniquercE. Mrstypl Ebederwto prMtyct the intygripyAand autherbypstyAof E.mersage; fotMexampCo, v erif)cs.ipn of a mersage Euthentipatyotbcstwr(MAC) or a digipy. EcgnatujM.gStanbw.ds for Esyptygraphic Eof)ware E.bwha.dwarerto perf)tm EnbrMption E.e wqpwry availo.no,ibut sucssrsfulooAujipg EncsypCyon tyterburerEerurityAmay be a chS.oonging pCMblem.gA sc pgTo errotMin EcAtem dwrignbEr exerujyotbcan aloow sujcersful E.ta.k;.gSumetimkr an E .werMa.y ca.bobny.n unerbrMptedwEnformatipn withoujyEwpMctyy ujdoing thSrenbr Mptipt.gSee, e.g., Eraff)c E.alysip, TEMPEST, otMTrojan hSrsc.DigTpy.oEigTatujM anb wenbrMptiotbmuscybeE.plopd to thSrciphertexeywhenbit EpcEsMatyrw(typica.loAEn Eh erscme devips ujcrwto compCte thermersage) to avotd EympCripg; Etherwipe anyAnotw betw qerbthe senderManbwEhSrerbrMptyon Egent csujd poterbialoy tamper withSEp. EnbsMpCing E. thSrtime of)creayptbipconly serure if thSrencsMptyotbEwvips ityerf has nbtybnenbtympe red wqph..

After applying second encryption step we obtain the following fragmentation key:

Table 12: The fragmentation key of the second message

(; (!,",#,$));(s; (%));(e;(&,',*,+));(t; (-,/,0));(I; (1,2)); (o; (3.4)); (l; 5)); (n; (6,7));(c;(8));(r; (9,:));(y ; (<));(a ; (=,>));(d; (?))

And the following encrypted message:

Table 13: Message deduced after the second step of encryption

E bsMpCyp6,ib<1-ycrf,!8s."p93/&rs#Eh'$84tf2pw6-y=51/yAEfm+%c>gTr,!bu0"3th&: #'rh72qu*rcE.Mr%/yp5Eb&?'rw04"p9Mty8-#/h+$160yg:2p<A=7?>u-h&rbyps0yA3f"E .m'r%=g*;$f4tM+x>mpCo,v&92f)8s.1p6!3f"=#m'r%>g*$Eu-h+7/2p=0y4tb8stwr(MA C)!3:">#?1g2py.$Ecg6=-ujM.gS/>7bw.?%f4:!Es<p0yg9=ph18"E3f)w>:'#E.bwh=.?w >9*r-4p+:f)tm!E6b9Mp/237"E.&#wqpwr<$=v>15o.no,ibu0%u8ssr%fu5ooAuj2pg"E 68s<pCy47#-yt&rbu9'rE*ru:1/yAm=<!b+">#8hS.oo6g27g$pCMb5&m.gA%cpgTo!* 9:4tM16#EcA0+m$?wr2g7bE9&x'rujy3tb8=6">5oow#%uj8*r%fu5$E./=.k;.gSum+ 01mkr>7!E.w&:M=.<"8>.b3bny.6$u7'rb9Mp-*?wE6f4:m=/1p7w20h3ujyEwpM8-y<" uj?416g#/hSr&7b9Mp02pt.gS'*,+.g.,!E:>ff)8"E.=5<%1p,#TEMPEST,$3tMT94j>7- hS:%c.D2gTpy.oE1gT=/ujM">6bw'7b9Mp023tbmu%cyb*E.p5op?!-4"/hSr81ph& :0'xeywh*6b2-EpcEsM=/yrw(0<p18>.5oAE7$Eh&r%cm'?*v2ps!ujcrw-3#84m pCt&$/h'rm*r%=g+)03!>v4t?"EympC91pg;#Eth&:w2p'$=6<A73twb+/wq&r b0h'"%*6?+rM>7bwEhSr&rb9Mp-y46Eg'7/!8suj?"p30*rb1=5o<#->mp+:$w2/hSEp.- E6bsMpC17g!E."/hSr02m&$4f)89'=.yptb1pc365<"%*ru:+#2f$-hSr&78sMp/y4tbEwv 1ps!20y+rf"h>%#6btybn&7b-ymp'9*?wqph..

Compared with IDEA and AES systems [9], [10], we noted that our system is extremely fast, either in encryption or decryption.

4. DISCUSSION

We will compare our encryption system in a well-known symmetric encryption system. This is IDEA. The latter is considered by specialists as one of the best private key crypto systems. It is used by PGP (Pretty Good Privacy) to encrypt data. The length of the key is 128 bits. As for our system the key is automatically generated and it is at least of the order of 30 * 8 = 240 bits (it can even go up to 256 * 8), because we can always assume that the number of different characters in the text is at least equal to 30 (if we can add it so that it is); even larger because more resistant to cryptanalysis by exhaustive search. In addition, this key is truly random whereas IDEA is not. IDEA operates on the plaintext, block by block; which is an asset to differential cryptanalysis, while our system acts on the plaintext in a comprehensive manner. So it is, firstly, away from such a cryptanalysis, on the other hand, it is less expensive. However, the only formidable cryptanalysis in our case is that based on the study of the frequency of occurrence of letters in the text. Now with change frequencies of appearance of characters, performed on the plaintext, and the introduction of new characters, thus ending any attempt of this kind of attack.

5. CONCLUSION

In this paper we have introduce a new symmetric ciphering system, based on evolutionary techniques. Our system possesses two keys, the genetic key and the fragment key. The genetic key is generated by the evolutionary algorithm, and whose size is very important. It resists much better to cryptanalysis via exhaustive search. The fragment key is generated by the fragmentation algorithm. Its application enhances the security of the system against attacks based on the study of frequencies. These keys are session keys and randomly generated by our system, Contrary to well-known crypto-system such as DES, IDEA or AES [9], [10].

Through this work, we reached two objectives. The first one is the introduction of new method of ciphering whose major advantage is strengthening the system against the most formidable attacks, as for the other advantages, they are mentioned above in the discussion. The second objective is to exploit the evolutionary algorithms in order to conceive and realize a ciphering system that benefits from all its qualities.

Despite the advantages of our system over other symmetric systems, we can't mathematically prove, nor claim, that our system is unconditionally secure, nor perfectly secure. Only Vernam cipher or on-time-pad is proved unconditionally secure. Shannon himself has demonstrated in the article [11].

REFERENCES

[1] F. Omary, A. Tragha, A. Bellaachia, A. Mouloudi "An Evolutionary Algorithm to Cryptography ". ICCMSE 2005

[2] Catherine KhamPhang. BOUNSAYTHIP "Algorithmes heuristiques et évolutionnistes" Thèse de doctorat université de Lille.Octobre 1988.

[3] Goldberg D.E. " Genetic algorithms in search optimisation & Machine Learning " (Addison-Wesley. Publishing Company, Inc) 1989.

[4] Mühlenbein H., "Evolutionary Algorithms: Theory and applications" (Wiley) 1993. Lecture Series on computer and Computational Sciences Vol.4, 2005, pp. 1749-1752

[5] F. Omary, A. Mouloudi, A. Tragha, A. Bellaachia, " A new ciphering method associated with evolutionary algorithm" December 2005. Accepted by ICCSA 2006 and will be published by Springer-Verlag Lecture Notes in Computer Science series (LNCS, SCIE).

[6] A. Mouloudi, F. Omary , A. Tragha, A. Bellaachia, “An Extension of Evolutionary Ciphering System”, 2006 International Conference on Hybrid Information Technology. November 9th ~ 11th, 2006, Korea
[7] Grenfenslette,j.j, "optimisation of control parameters for genetic algorithms", IEEE translation on systems Man, and cybernetics, Vol 16 N°1 pp122-128.1986.
[8] Christophe Caux, Henri Pierreval, Marie-Claude Portmann, " Les algorithmes génétiques et leur application aux problèmes d’ordonnancement ", APII. Volume 29-N° 4-5/1995, pp. 409 - 443.
[9] Alfred J.Menzes , Paul C. Van Dorschot , Scott A. Vanstone, Handbook of Applied Cryptography CRC Press fifth Printing (August 2001)
[10] Douglas R. Stinson, Cryptography – Theory and Practice, CRC Press 1995 ISBN 0-8493-8521-0
[11] Claude Shannon, “ Communication Theory of Secrecy Systems”, Bell Systems Technical Journal (1949).

AUTHORS

A. MOULOUDI was born in 1959 in Morocco. He received the B.S degree in applied mathematics from Mohamed V University in Morocco at 1982; Master in Computer Science from the University Mohammed V, at 1984; Ph.D in Computer Science from the same university, at 1988. He obtains Habilitation to direct academic research in Computer Science, from Ibn Tofail University in Morocco, at 2008. Currently, A. MOULOUDI is a member of the laboratory MISC (Information Modeling and Communication System), at the sciences faculty, Ibn Tofail University, in Morocco.