International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org
Volume 4, Issue 9, September 2015 **ISSN 2319 - 4847**

Hybrid encryption solution for securing data

Stelian Dumitra[1], Bogdan Gabriel Vasilciuc[2]

[1]Department of Economic Informatics Doctoral School, The Bucharest University of Economic Studies, Bucharest, Romania

[2]Department of Economic Informatics Doctoral School, The Bucharest University of Economic Studies, Bucharest, Romania

ABSTRACT

The dynamic of change from the current business environment has forced organizations to adopt some of the most complex technological solutions. Globalization, virtualization of environments and the transfer of large amounts of operational and strategically data present important challenges when it comes to business processes, especially from a security point of view. On one hand, the technological progress can bring many advantages. On the other hand, the organizations must pay special attention to threats and attacks coming from outside their system, which, in the end, translate to loss of capital. The following research paper analyzes the most important cryptography solutions, which can guarantee security at the information level, and proposes a hybrid algorithm, which ensures confidentiality, integrity and data authenticity.

Keywords: Cryptography, Hash function, Message Authentication Code, Hybrid encryption, AES, RSA

1. INTRODUCTION

Encryption, as a component of a larger field, ***information security***, focuses on a series of general objectives:

- **Authentication** – the propriety to confirm an identity attributed to an entity or a user.
- **Confidentiality** – the propriety to forbid unauthorized access to people for whom the information is not destined.
- **Data integrity** – the propriety that entails that the transmitted (saved) data cannot be altered or modified by anyone other than the authorized persons, which means that the message received during a transaction coincides with the message originally sent.
- **Non-repudiation** – the propriety to prevent the negation of a series of initial transactions (phenomena, events), especially by the expeditor, thus guaranteeing the integrity and origin of the transactions.

ACCORDING TO DOUGLAS STINSON [1], "AN ENCRYPTION SYSTEM REPRESENTS THE QUINTUPLE (P, C, K, E, D) FOR WHICH THE

following conditions are satisfied:

- P is finite set of clear texts;
- C is a finite set of encrypted texts;
- K, the key space, is a finite set of keys;
- Each $k \in K$ determines an encryption method $e_k \in E$ and a correspondent decryption method $d_k \in D$. Each $e_k : P \to C$ and $d_k : C \to P$ is a function which satisfies the following property: $d_k\left(e_k(x)\right) = x, \forall x \in P$"

Dependent on the type of encryption/decryption keys used, encryption systems are divided into two major categories [2], [3]:

- **Symmetrical or secret keys encryption systems** use the same key for both the encryption and decryption of the message ($k_e = k_d = k, k \in K$). Generally they are used to ensure privacy of the data.
- **Asymmetrical or public keys encryption systems** use different keys for encryption and decryption respectively ($k_e \neq k_d, k_e, k_d \in K$). The encryption key of an entity is made public and the decryption one is kept secret, as it is known only by that entity. Generally they are used in the transportation (distribution) of the symmetrical encryption keys and in implementing digital signatures.

2. COMPARISON BETWEEN ENCRYPTION SYSTEMS WITH SYMMETRICAL AND ASYMMETRICAL KEYS

In order to ensure data confidentiality, it is recommended to use symmetrical encryption algorithms, as they have high levels of performance in terms of processing time, allowing for the fast encryption of messages of different sizes. The security of symmetrical algorithms depends on the type of key management used in the encryption. The main disadvantages of symmetrical keys encryption systems are the following:

- The encryption key needs to be permanently kept secret in two distinct places;

- The larger the text, the easier it is to break;
- In large networks, key management becomes a very difficult problem;
- A secure communication channel is needed for key transmission.

The main use for the public keys encryption system is in ensuring the confidentiality of the messages and ensuring the authenticity (digital signature). Among the applications of the public keys encryption systems are:

- The transportation (distribution) of symmetrical encryption keys;
- The implementation of a series of digital signatures schemes;
- Protocols for authentication of entities.

The advantages of using public keys encryption systems are:

- They always need a pair of keys, one public and one private;
- The public key can be distributed through any other channel of communication (non-secure), and the private key needs to kept secret;
- Encryption is done with one of the keys and decryption is always done with the other;
- The management of the keys can be attributed to one entity (server, "trusted authority");
- The pair of keys can be used for a long period of time;
- In a distributed system with *n* users the total number of keys used in the secure communication between pairs is *2n*, that is *O(n)*, while for a symmetrical system there is need for:

$$1 + 2 + 3 + \cdots + (n-1) = \frac{n(n-1)}{2} \ (1) \text{ keys, that is } O(n^2).$$

The main disadvantages of public keys encryption systems are:

- They are significantly slower that symmetrical encryption systems;
- Much longer keys are needed;
- In order to ensure the security of a public key algorithm problems are used that are very hard to solve through number theory (factorizing of very large integer numbers, calculation of discrete logarithms);
- The absolute security of a public key encryption scheme cannot be guaranteed;
- They have only been in use since the end of the 70s and thus lack experience.

In this article we will introduce a new encryption solution, a combination of the current encryption methods: symmetrical encryption, asymmetrical encryption, hash functions, MACs, solution that ensures privacy, integrity and non-disclaimer of data. It can be integrated as a component in any software solution for data encryption.

As it was previously stated, encryption has a series of general objectives, including: authentication, confidentiality, data integrity and non-repudiation.

In order to increase data security, current encryption systems can be combined and new hybrid encryption systems created. A general issue emerges: "what components are necessary in a hybrid encryption system in order to secure private data that are exchanged in a vulnerable network like the internet?" Generally, these components are:

- Symmetrical encryption algorithms for ensuring data privacy (DES, AES, 3DES, RC4);
- Session keys generators ((pseudo) random generators);
- Session keys used by symmetrical algorithms;
- Infrastructure of public keys (creation, organizing, storing, distributing, revoking);
- Terminal private keys used for encryption of session keys;
- Master keys for encryption of terminal secret keys;
- Asymmetrical encryption algorithms used for encryption/decryption of session keys (RSA, El – Gamal);
- Hash encryption functions for ensuring the integrity of exchanged data;
- Digital signatures for ensuring the non-repudiation of the data.

In the following section we will introduce a series of aspects regarding the security of current encryption systems (vulnerabilities, advantages, disadvantages).

3.SECURITY OF CURRENT ENCRYPTION SYSTEMS

The main attacks on encryption systems are:

- Attempts to obtain the clear text in the encrypted data sent;
- Attempts to find out the key used, knowing the clear texts and the encrypted message.

3.1 Security of Symmetrical Encryption Systems

The DES system, on 56 bits, was hacked in July 1998 by the American organization *Electronic Frontier Foundation* in 3 days, with the help of a special computer. A series of controversies regarding DES were investigated by the encryption systems, most of them related to its design [4, 5]. At present it exists under the name of 3DES. This system has an advantage over the DES system because the length of the keys grows from 56 bits (64 including the parity bits) to 168 (192 including the parity bits), which leads to the growth of the key space to 2^{168}. Thus, the system becomes much stronger in case of a *meet-in-the-middle* attack. The most popular attack methods on symmetrical keys systems are:

- **Brute force attack** – testing all the possible variants in the key space, until the right key is discovered. For example, if an encryption algorithms used a 128/192/256 bits key, then the maximum number of keys on which the testing is done is $2^{128}/2^{192}/2^{256}$, an extremely large number. This attack method is efficient when the encryption key length is small and the encryption space is small.
- **Encrypted text attack** – the type of attack in which the cryptanalyst only has the encrypted text and based on it he will need to identify the clear text and/or the encryption algorithm and/or encryption key. It is the hardest type of attack, and the algorithms that do not resist such an attack are considered completely insecure.
- **Attack with a clear known text** – the type of attack in which the cryptanalyst has one or more pairs of (clear text, encrypted text) and based on this information he will need to identify, through specific methods, the encryption algorithm and/or key.
- **Attack based on chosen clear text** – the type of attack through which the encryption analyst has the possibility to chose the clear text and based on it, find out the encryption algorithm and/or key.
- **Linear cryptanalysis** – a type of cryptanalysis introduced by Matsui and Yamagishi in 1991. It is a type of attack with known clear text. This technique is applied to bloc numerical combinations and flux numerical combinations. Fist it was applied to the FEAL encryption system [6] and then to the DES system, attack which is this case needs 2^{47} clear known texts [7]. The basic principle of the linear cryptanalysis is establish is a linear relationships between a subset of bits from the clear text, the bits in the encrypted texts and the bits of the sub-keys.
- **Differential cryptanalysis** –was proposed in 1991 by Eli Biham and Adi Shamir [8, 9]. The technique is a part of category of the chosen clear text attacks.

At present the most efficient attack is the brute force one for AES. The brute force attack needs: 2^{127}Rijndael encryptions for a 128 bits key; 2^{191}Rijndael encryptions for a 192 bits key; 2^{255} Rijndael encryption for a 256 bits key.

A series of reference works regarding AES security can be found at [10, 11, 12].

In order to ensure an increased level of security of an encryption text, the special publication NIST 800-38A (Dworkin, 2001) has made recommendations regarding the operating modes of symmetrical encryption, how the transformation (block code) is applied on the message blocks [13]. More operating modes were recommended, among which the most important are: ECB (Electronic Codebook Mode), CBC (Cipher Block Chaining), CFB (Cipher Feedback Mode), OFB (Output Feedback Mode), CTR (Counter Mode). In the CBC and CFB, the modification of a clear text P_i generates modifications in all texts of encryption blocks P_i.

3.2 Security of Asymmetrical Encryption Systems

Below we enumerate a few attacks on RSA which can be found at [14]:

- Knowing the Euler indicator is enough for hacking the RSA system;
- Weiner's attack is an attack on the decryption exponent *d*, small in size and the public key *e* is smaller than the absolute value *n*;
- Attack on public exponents of small sizes;
- Factorizing the absolute value knowing the decryption exponent;
- Attack with chosen encrypted text;
- Attack through partial exposure of public/private key;
- Attack through partial exposure of prime factors etc.

3.3 Security of Hash Encryption Functions

In 2004 collision type attacks on MD5 and SHA-0 functions were demonstrated by the Chinese researchers Xiaoyun Wang and Hongbo Yu [15]. A year later, Wang extended a theoretical attack on SHA-1, maintaining that a collision can be found in approximately 2^{69} evaluations [16]. An enhanced version of this attack was proposed by Wang in August 2005, the complexity time being approximately 2^{63}, the number of operations necessary for a *brute-force* attack being 2^{80}) [17]. In 2005 Vlastimil Klima discovered a more rapid solution than Wang for finding collisions for MD5 [18]. Other encryption attacks on SHA-1 were proposed by Christophe De Cannière and Christian Rechberger

[19], Grechnikov [20], Stéphane Manuel [21], Cameron McDonald, Philip Hawkes and Josef Pieprzyk [22], Marc Stevens [23].

4 HYBRID ENCRYPTION SOLUTION

In order to increase data security and because of the disadvantages of current encryption systems (key management, processing speed, vulnerabilities) we propose a hybrid encryption solution which ensures the confidentiality, integrity and authenticity of the data. It is used as a software component (library) on the .NET platform in the systems for electronic payments processing and, also, in can be used in any Microsoft application.
In this encryption solution we have used the following encryption components:

- As a symmetrical key encryption algorithm I used AES (Rijndael) in the CBC mode, and the IV generated randomly. We have chosen AES because it resists current cryptanalytic attacks, it offers more security than DES and 3DES, it was a very strong and simple design (it is based on algebraic operations from *GF(256)*) and it has a high encryption speed on most operation platforms (software and hardware).
- The keys used in the symmetrical encryption are generated based on RNGs. These symmetrical keys are encrypted using the RSA algorithms and stored in a database.
- The RSA keys used in the encryption/decryption of symmetrical keys are stored in the operating system's store, along with the entity certificates, and for safety they are also saved in a data base.
- Since the encryption/decryption processes happen on the same machine, we have replaced digital signatures with MAC, a lot more efficient. These are different than digital signatures in that they can be used by both parties using the same secret key, while digital signatures use different keys for singing and verifying. In this case we used the HMACSHA512 function.
- The keys used for calculating the MAC are generated, encrypted and stored in the same way as the symmetrical keys.
- Based on the length of the symmetrical key (whose length is measurable 128, 192, 256 bits), during the encryption process, in order to build a strong symmetrical keys we used the hash algorithms: MD5, SHA-256, SHA-384.

The general outline of the solution can be found in the diagram below:

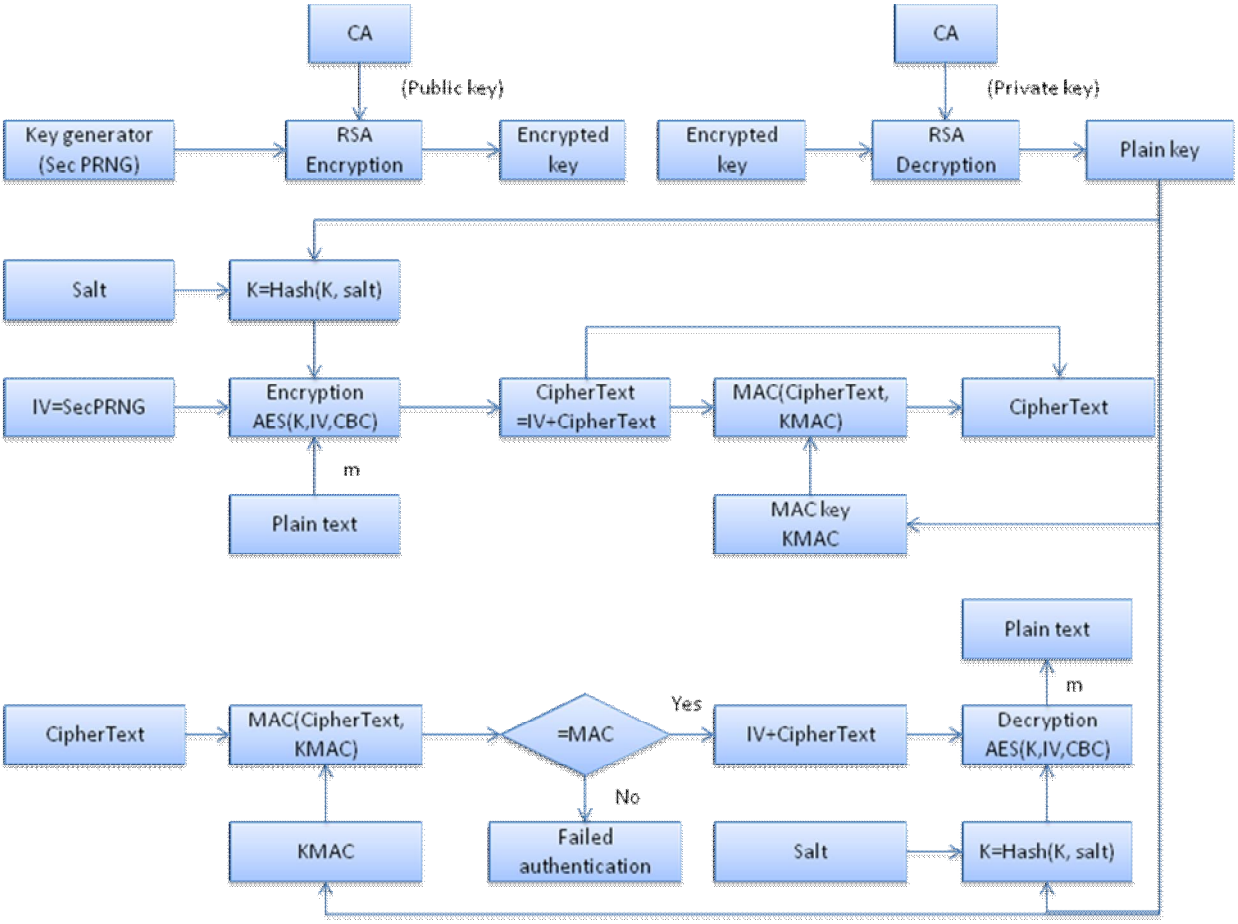

Figure 1 Hybrid encryption scheme

The process for the proposed solution is as follows:

1) Symmetrical keys, both for symmetrical encryption/decryption and for calculating the MAC are generated based on RNGs.
2) These keys are in turn publically encrypted with a series of keys issued by a certifying authority and stored in the operating system's store.
3) The final key which enters the encryption/decryption system is hashed based on the standard hash algorithms: SHA256, SHA384, respectively MD5. In order to increase security, these keys are in turn combined with some constants (leaps), according to the relation K = h(K||Salt), where h(x) = MD5(x) for AES-128, h(x) = SHA384(x) for AES-192, respectively h(x) = SHA256(x) for AES-256.
4) The initialising vector IV, in order to be used in decryption, is attached to the final encrypted text, so that: CipherText = IV || CipherText.
5) The MAC of the final encrypted texts is calculated based on the relation: MAC = HMACSHA512 (HMACSHA512 (IV || CipherText, KMAC), KMAC).
6) The MAC is stored in the data base, together with the encrypted text EncryptedText = IV + CipherText.
7) In the decryption, the symmetrical key and the MAC key respectively, are decrypted with the private key RSA.
8) The authenticity of the encrypted message is verified and if the MACs are the same then the message was not altered.
9) The encrypted message is decrypted using the symmetrical key built in the same way as it was built in encryption, K = h(K || Salt).

5. BENEFITS OF THE HYBRID SOLUTION

The proposed encryption method brings a series of benefits, among which we enumerate:

- The privacy is ensured, using the symmetrical encryption algorithm AES in the CBC mode and IV generated based on RNG. Also, the protection against cryptanalytic attacks on symmetrical systems is ensured because of its design.
- The encryption speed is high, because of AES.
- The symmetrical keys and the MAC keys are generated randomly using secured RNGs from .NET.
- The protection of symmetrical keys/MAC, as they are encrypted on the basis of RSA.
- The RSA keys are issued by a certifying authority and stored in the operating system's store and in the data base.
- Data integrity is ensured by using hash encryption functions.
- Authenticity and, again, integrity is ensured by calculating the MAC of the final encrypted text.

Moreover, the hybrid encryption method has solved a few problems of the current encryption systems:

- The management of symmetrical keys and the ones used in calculating the MAC in terms of generating and storing.
- The simultaneous satisfaction of multiple security requirements: privacy, integrity and authenticity.
- The use of powerful keys in the encryption/decryption process (through combining them with other current methods- hash functions).

6. CONCLUSIONS

The proposed hybrid method ensures confidentiality, integrity and authentication of data, providing the certainty that the data was not modified as long as it was stored in the data base. The security is also increased due to the fact that the AES algorithm is used in the CBC mode and based on an IV generated randomly.

REFERENCES

[1] D. Stinson, Cryptography, Theory and Practice, Third Edition, Ed Chapman & Hall/CRC, Boca Raton, 2006, ISBN: 1584885084.
[2] I. Ivan, C. Toma, Informatics Security Hand book - 2nd Edition, ASE Publishing House, 2009, ISBN 9786065052468.
[3] A.J. Menezes, P.C. van Oorschot, S. Vanstone, Handbook of Applied Cryptography, Ed CRC Press, 1996, ISBN: 9780849385230
[4] K. Nyberg, "On the Construction of Highly Nonlinear Permutations", Advances in Cryptology – EUROCRYPT '92, Springer-Verlag (New York, NY), pp. 92–98, 1993
[5] J. Seberry, X-M. Zheng, "Highly Nonlinear 0-1 Balanced Boolean Functions Satisfying Strict Avalanche Criterion", Advances in Cryptology – AUSCRYPT '92, Springer-Verlag (New York, NY), 1993, pp. 145–155.
[6] M. Matsui, A. Yamagishi, "A New Method for Known Plaintext Attack of FEAL Cipher", Advances in Cryptology — EUROCRYPT' 92, Lecture Notes in Computer Science, Springer, vol. 658, pp 81-91, 1993
[7] M. Matsui, "Linear Cryptanalysis Method for DES Cipher", Advances in Cryptology – EUROCRYPT '93, Springer-Verlag (New York, NY), 1994, pp. 386–397

[8] E. Biham, A. Shamir, "Differential Cryptanalysis of DES-like Cryptosystems", Journal of Cryptology, 4, pp. 3–72, 1991
[9] E. Biham, A. Shamir, "Differential Cryptanalysis of the Data Encryption Standard", Springer-Verlag (New York, NY), 1993
[10] T. Tiessen, L.R. Knudsen, S. Kölbl, M.M. Lauridsen, "Security of the AES with a Secret S-box", Cryptology ePrint Archive, Report 2015/144, [Online]. Available: http://eprint.iacr.org/2015/144.pdf, [Accessed: Mai 10, 2015].
[11] A. Bogdanov, D. Chang, M. Ghosh, S.K. Sanadhya, "Bicliques with Minimal Data and Time Complexity for AES (Extended Version)", Cryptology ePrint Archive, Report 2014/932, [Online]. Available: http://eprint.iacr.org/2014/932.pdf, [Accessed: Mai 10, 2015]
[12] A. Biryukov, O. Dunkelman, N. Keller, D. Khovratovich, A. Shamir, "Key recovery attacks of practical complexity on AES-256 variants with up to 10 rounds", EUROCRYPT'10, vol. 6110 of LNCS, Springer, pp. 299 - 319, 2010
[13] National Institute of Standards and Technology (NIST), "Recommendation for Block Cipher Modes of Operation, NIST PUB 800-38A", 2001, [Online]. Available: http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf , [Accessed: Mai 10, 2015]
[14] S. Yan, "Cryptanalytic Attacks on RSA", Springer Verlag, 2008, ISBN: 0387487417
[15] X. Wang, H. Yu, "How to Break MD5 and Other Hash Functions", Advances in Cryptology – EUROCRYPT 2005, LNCS, vol. 3494, pp 19-35, Springer, Berlin Heidelberg, 2005
[16] X. Wang, Y.L. Yin, H. Yu, "Finding Collisions in the Full SHA-1", CRYPTO 2005, LNCS, vol. 3621, pp 17-36. Springer, Berlin Heidelberg, 2005
[17] X. Wang, A. Yao, F. Yao, "Cryptanalysis of SHA-1", Cryptographic Hash Workshop hosted, NIST, October 2005
[18] V. Klima, "Finding MD5 Collisions – a Toy For a Notebook", Cryptology ePrint Archive, Report 2005/075, [Online]. Available: http://eprint.iacr.org/2005/075.pdf, [Accessed: Mai 11, 2015]
[19] C. De Cannière, C. Rechberger, "Finding SHA-1 Characteristics: General Results and Applications", Advances in Cryptology – ASIACRYPT 2006, LNCS, vol. 4284, pp 1-20, Springer, Berlin Heidelberg, 2006
[20] E.A. Grechnikov, "Collisions for 72-step and 73-step SHA-1: Improvements in the Method of Characteristics", Cryptology ePrint Archive: Report 2010/413, 2010
[21] S. Manuel, "Classification and generation of disturbance vectors for collision attacks against SHA-1", Designs, Codes and Cryptography, vol. 59, Issue 1-3, pp 247-263, Springer, US, 2011
[22] C. McDonald, P. Hawkes, J. Pieprzyk, "Constructing Nonlinear Differentials in SHA-1", Cryptology ePrint Archive: Report 2009/259, 2009
[23] M. Stevens, "Counter-Cryptanalysis", Advances in Cryptology – CRYPTO 2013, LNCS, vol. 8042, pp 129-146, Springer, Berlin Heidelberg, 2013

AUTHOR

Stelian Dumitra graduated Economic Informatics Doctoral School, at Bucharest University of Economic Studies in September 2015. In present he works as Senior Software Engineer at Endava Romania. Currently, his research interests include: Information Security, Cryptography, Web Services, Mathematics, and Software Development.

Bogdan Gabriel Vasilciuc graduated Polytechnic University of Bucharest specialization in Economic Engineering and got a master degree in "Industrial business management in the context of EU integration" within the Department of Management at UPB. Currently, he is a PhD student in the field of Economic Informatics at the Academy of Economic Studies. He is also currently Vice Dean and lecturer at Economic Science Faculty, Titu Maiorescu University of Bucharest where he teaches courses "Databases" and "E-Commerce". Subjects of his current interest are databases, document management, content management, information management, business process management and enterprise 2.0.