International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:3, No:6, 2009

Software to Encrypt Messages Using Public-Key Cryptography

E. Inzunza-González, C. Cruz-Hernández, R. M. López-Gutiérrez, E. E. García-Guerrero, L. Cardoza-Avendaño, H. Serrano-Guerrero

***Abstract*—**In this paper the development of a software to encrypt messages with asymmetric cryptography is presented. In particular, is used the RSA (**R**ivest, **S**hamir and **A**dleman) algorithm to encrypt alphanumeric information. The software allows to generate different public keys from two prime numbers provided by the user, the user must then select a public-key to generate the corresponding private-key. To encrypt the information, the user must provide the public-key of the recipient as well as the message to be encrypted. The generated ciphertext can be sent through an insecure channel, so that would be very difficult to be interpreted by an intruder or attacker. At the end of the communication, the recipient can decrypt the original message if provide his/her public-key and his/her corresponding private-key.

***Keywords*—**Asymmetric cryptography, Prime number, Public-key, Private-key, Software**.**

I. Introduction

PUBLIC-KEY cryptography is a method for secret communication between two parties without requiring an initial exhange of secret keys as well as symmetric cryptography also known as secret-key cryptography, this kind of cryptography uses a single secret key for both encryption and decryption. A core problem of the symmetric cryptosystems is key distribution and key management [4]. For the key-exhange, they need, for example, a secure channel or a courier. Figure 1 shows a block diagram of the symmetric key cryptography [3]. The key exchange problem becomes even more difficult if many people want to exchange encrypted messages, for example on the internet. If a network communication system has n users and any two of them exchange a key, then n(n-1)/2 secret key exchanges are necessary and all those keys have to be stored securely [4].

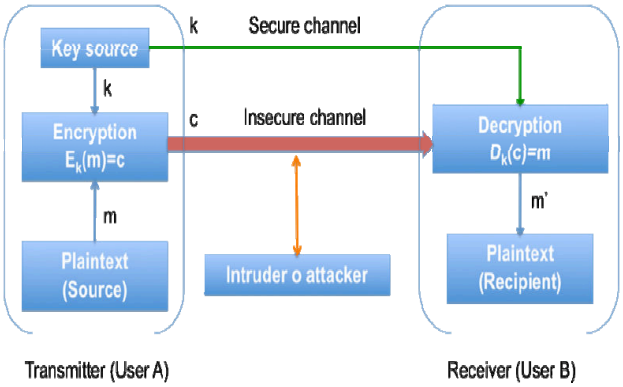

Fig. 1 Block diagram of the communication system using symmetric key cryptography [3].

Public-key cryptography is a fundamental and widely used technology around the world, and enables secure transmission of information on the internet and other communication systems; this concept was proposed in [7]. It is also known as *asymmetric cryptography* because the key used to encrypt a message differs from the used to decrypt it. In public-key cryptography, a user has a pair of cryptographic keys – a public-key and a private-key. The private-key is kept secret, while the public-key may be widely distributed and known for any user. Messages are encrypted with the recipient's public-key and can only be decrypted with the corresponding private-key, see Fig. 2 [3]. The keys are related mathematically, but the private-key cannot be practically derived from the public-key. In addition, the public-key cryptosystems can generate digital signatures and can be combinated with symmetric cryptosystems [1] and [6].

E. Inzunza is with the Baja California Autonomous University (UABC), Ensenada, B.C. 22860 México (corresponding author to provide phone: +52 646-175-0744; fax: +52 646-174-4333; (e-mail: einzunza@ uabc.mx).

C. Cruz Hernández is with the Electronics and Telecommunications Department, Scientific Research and Advanced Studies of Ensenada (CICESE), Ensenada B.C. 22860 México (corresponding author to provide phone: +52-646-175-0500; fax: +52-646-175-0500; (e-mail: ccruz@cicese.mx).

R. M. López Gutiérrez is with the Baja California Autonomous University (UABC), Ensenada, B.C. 22860 México (corresponding author to provide phone: +52 646-175-0744; fax: +52 646-174-4333; (e-mail: roslopez@uabc.mx).

L. Cardoza Avendaño is with the Baja California Autonomous University (UABC), Ensenada, B.C. 22860 México (corresponding author to provide phone: +52 646-175-0744; fax: +52 646-174-4333; (e-mail: lcardoza@uabc.mx).

E. E. García Guerrero is with the Baja California Autonomous University (UABC), Ensenada, B.C. 22860 México (corresponding author to provide phone: +52 646-175-0744; fax: +52 646-174-4333; (e-mail: eegarcia@uabc.mx).

H. Serrano is with the Baja California Autonomous University (UABC), Ensenada, B.C. 22860 México (corresponding author to provide phone: +52 646-175-0744; fax: +52 646-174-4333; (e-mail: hazael@uabc.mx).

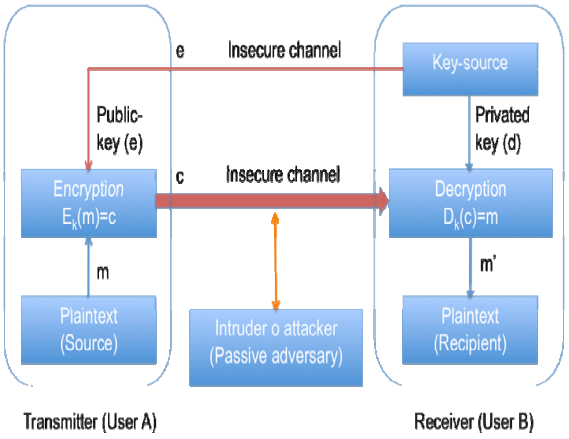

Fig. 2 Block diagram of the communication system using asymmetric cryptography [3].

The aim of this paper is the development of a software to encrypt alphanumeric information using the RSA algorithm described in [1], was the first public-key cryptosystem and is still the most important. Its security is closely related to the difficulty of finding the factorization of a composite positive integer that is the product of two large number primes [1].

II. RSA Cryptosystem

A. Key Generation [1]

We explain how the recipient's (in the future we will say user B) generates his private and public RSA keys. User B generates randomly and independently two large prime numbers p and q and computes the product

$$n = p \cdot q \tag{1}$$

The numbers p and q of (1) must be large so that it is not computationally feasible for anyone to factorize n. (Remember that n, but not p and q, will be in the public file), the authors [1] recommend the use of numbers with 100 decimal digits, so that n has 200 digits. Next we need to calculate φ(n) as follows [1]:

$$\varphi(n) = (p-1)\cdot(q-1) \tag{2}$$

Then the user B chooses an integer *e* with

$$1 < e < \varphi(n) \quad and \quad \gcd(e, \varphi(n)) = 1 \tag{3}$$

where *gcd* is the greatest common divisor of *e* and φ(n).

Note that *e* is always odd since (p-1) is even. User B computes an integer *d* with

$$1 < d < \varphi(n) \quad and \quad d \cdot e \equiv 1 \bmod(\varphi(n)) \tag{4}$$

Since *gcd*(*e*, φ(n)) = 1, such a number *d* exists. It can be computed by using the Extended Euclidean Algorithm [8].
The *public-key* of the user B is the pair (*n,e*) and his *private-key* is *d*. The number *n* is called the RSA modulus, *e* is called the encryption exponent and *d* is called the decryption exponent.

B. Encryption [1]

To encrypt a message *m* with RSA algorithm, using a *public-key* (*e, n*), proceed as follows. (Here *e* and *n* are a pair of positive integers). First, represent the message as an integer between 0 and (n−1), i.e.,

$$0 \le m < n \tag{5}$$

Break a long message into a series of blocks, and represent each block as such an integer. Use any standard representation.
A plaintext *m* is encrypted by computing

$$c = m^e \bmod(n) \tag{6}$$

The ciphertext is c. Then if the user A knows the public-key (*n, e*) of user B, the user A can encrypt messages to be sending to user B. to make encryption efficient, use fast exponentiation.

C. Decryption [1]

The decryption of RSA algorithm is based on the following theorem.

Theorem 1 [4]. Let (*n,e,*) be a public RSA key and *d* the corresponding private RSA key, then

$$m = c^d \bmod(n) \tag{7}$$

Similarly, the decryption key is the pair of positive integers *d* and *n*.

III. Software Development

First, we develop a graphical user interface with MATLAB R2008b® (see Fig. 3) to generate the keys according to the methodology described in the previous section. As seen in Fig. 3, the user must provide the prime numbers p and q, then press the button **Generate Public-Key**. Then, the program checks that p and q are prime numbers. While p and/or q are not prime numbers, the program displays the warning window shown in Fig. 4 and 5. If p and q are primes, the program calculates n, φ(n), and the number of public keys generated. After the user B has to choose randomly a public-key (e).

Finally the user B must press the button **Generate Private-Key** to generate the corresponding private-key.

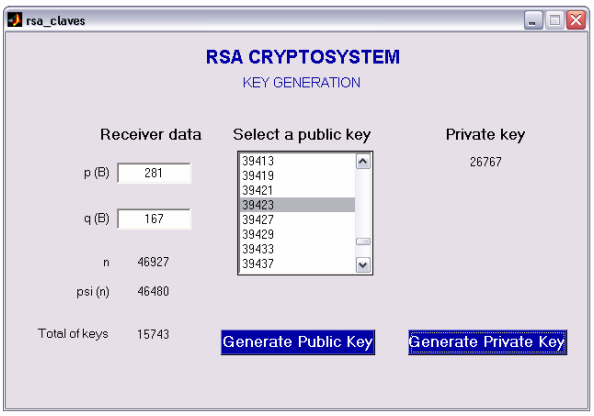

Fig. 3 Graphical user interface for generating public and private keys.

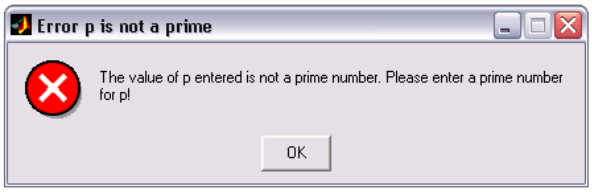

Fig. 4 Warning window if p is not a prime number.

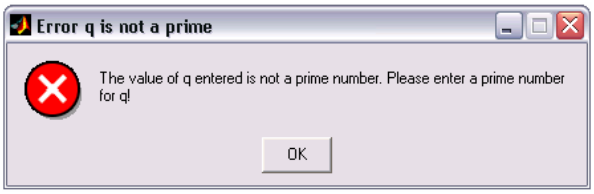

Fig. 5 Warning window if q is not a prime number.

Secondly, we develop the graphical user interface to encrypt messages. As shown in Fig. 6, to encrypt a message *m* with RSA algorithm, the user A must enters a public-key (*e, n*) of the user B, then the user A enters the message to be encrypted and sent to user B, and then the user A must press the button to encrypt. Finally, the program generates the ciphertext (c). This ciphertext can be sent through an insecure channel from user A to user B.

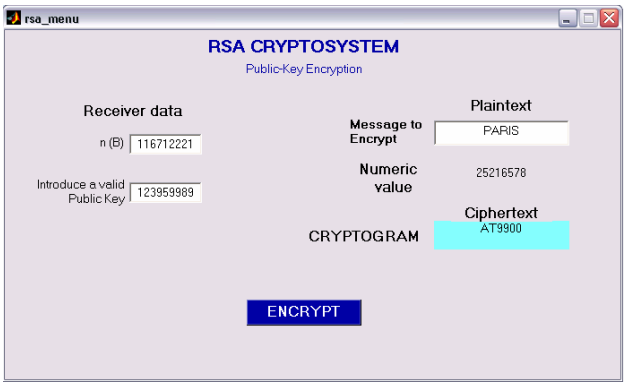

Fig. 6 Graphical user interface to encrypt messages using public-key cryptography.

In the sequel, we develop a graphical user interface to decrypt messages by using RSA algorithm. As shown in Fig. 7, to decrypt a message *m* with RSA algorithm, the user B (recipient's) must enters his public-key (n) and his private-key (d), then the user B enters the ciphertext to be decrypted, later the user B must press the button to decrypt. Finally, the program retrieves the plaintext (m'). This plaintext can be read clearly by the user B.

Fig. 7 Graphical user interface to decrypt messages using public-key cryptography.

IV. Results

In this section, we present some examples of encryption of alphanumeric information, from how to generate the public and private keys, and finally how to encrypt and decrypt by using the proposed software. Table I shows this example. For the encoding/decoding of alphanumeric information, we use the equivalence shown in Table II and III. Therefore, the message (m) must be represented as a number in base 36.

TABLE I
EXAMPLES OF DIFFERENT GENERATION OF PUBLIC-KEY WITH ITS CORRESPONDING PRIVATE-KEY WITH MESSAGES TO ENCRYPT, MESSAGES ENCRYPTED (CIPHERTEXT) AND RETRIEVED MESSAGE

Prime (p)	Prime (q)	n	φ(n)	Public-key (*e*) (selected ramdomly)	Private - key (*d*)	Message (m)	Ciphertext (c)	Retrieved Message (m')
281	167	46927	46480	39423	26767	YES	AG6J	YES
281	167	46927	46480	39423	26767	NO	XSG	NO
11317	10313	116712221	116690592	123959	46807463	2009	PI48B	2009
11317	10313	116712221	116690592	123959	46807463	12345	AH08VY	12345
11317	10313	116712221	116690592	123959989	84999325	PARIS	AT9900	PARIS
11317	10313	116712221	116690592	123959989	84999325	CESSE	A2XEHG	CESSE

TABLE II
CHARACTER ENCODING OF THE ENGLISH ALPHABET

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

TABLE III
CHARACTER ENCODING OF THE DIGITS 0 TO 9

0	1	2	3	4	5	6	7	8	9
26	27	28	29	30	31	32	33	34	35

The presented software is with purposes of illustration only, by using the RSA algorithm. Nevertheless, the developed software can be easily used:

1. With other encryption algorithms that work based on public-key cryptography; such as: Rabin, ElGamal, Generalized ElGamal, McEliece, Knapsack, Merkle-Hellman knapsack, Chor-Rivest napsack, Probabilistic, Goldwasser-Micali probabilistic, Blum-Goldwasser probabilistic [3].

2. In combination with nonlinear functions, for more details see [10,11].

3. With chaotic encryption, see e.g. [12-15].

V. CONCLUSIONS

In this paper, we have presented the development of software to encrypt messages with the RSA algorithm. This software can be used in universities and research centers as a tool for studying public-key cryptography. The program is friendly and easy to operate for users. Messages that are encrypted are English alphabet characters and digits 0 to 9. When you want to encrypt long messages must provide very large prime numbers or use RSA encryption as a block cipher [4]. For the generation of keys, the software first verifies that the numbers p and q are primes. If p and q are large prime numbers, the program can generate millions of public-keys and its corresponding private-keys. Then the user selects randomly the public-key and calculates the corresponding private-key. Now the user can easily encrypt alphanumeric information and send it through a public channel or media insecure. Only the user who has the private-key can decrypt the messages.

REFERENCES

[1] Rivest, R., Shamir A., Adleman L., A method for obtaining digital signatures and public key cryptosystems, Communications of the ACM. 21 (1978), pp. 120-126.
[2] B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, Wiley & Sons, Inc. 1996.
[3] A.J. Menezes, P.C. Van Oorschot y S.A. Vanstone, 1997. *"Handbook of Applied Cryptography"*, CRC Press. pp 15-28 and. 283-291.
[4] J. A. Buchmann, Introduction to Cryptography. Marietta, GA: Springer-Verlag, 2000, pp. 139-153.
[5] D.R. Stinson, *Cryptography: Theory and Practice*, Tercera edición, CRC Press. 2005.
[6] A. Fuster, *Técnicas criptográficas de protección de datos*, Ed. Ra-Ma, 2001.
[7] Diffie, W., and Hellman, M. *New directions in cryptography.* IEEE Trans. Inform. Theory IT-22, Nov. 1976, 644-654.
M. Bellare, P. Rogaway, Introduction to Modern Cryptography, San Diego, CA., 2005. pp. 211-230.
[8] V. Zur Gathen, Joachim; Gerhard, Jürgen*, "The Euclidean Algorithm", en Modern Computer Algebra.* Cambride University Press, (*2003),* ISBN 0-521-82646-2.
[9] W. Stallings*, Cryptography and Network Security: Principles and Practice,* 2da. Edición, Prentice-Hall, New Jersey, 1999.
[10] Yang T. Wu C.W. and Chua L.O., *Cryptography based on chaotic systems.* IEEE Trans. Circuits Syst. I 44(5)(1997)469–472.
[11] Cruz-Hernández C. and Serrano-Guerrero H*., Cryptosystems based on synchronized Chua's circuits.* In Procs. of the 16th IFAC World Congress, July 3–8(2005) Prague, Czech Republic.
[12] R.M. López-Gutiérrez, C. Cruz-Hernández, C. Posadas-Castillo, and E.E.García-Guerrero*, Encrypted audio transmission using synchronized*

Nd:YAG lasers, Proceedings Of World Academy Of Science, Engineering And Technology Vol. 30 (2008), ISSN 1307-6884.

[13] C. Posadas-Castillo, R. M. López-Gutiérrez, C. Cruz-Hernández, *Synchronization of chaotic solid-state Nd:YAG lasers: Application to secure communication,* Comm. in Nonlinear Science and Numerical Simulation 13, (2008), pp. 1655-1667.

[14] A.Y. Aguilar-Bustos a,b, C. Cruz-Hernández, *Synchronization of discrete-time hyperchaotic systems: An application in communications,* Chaos, Solitons and Fractals, In press doi:10.1016/j.chaos.2008.05.012.

[15] L. Gámez-Guzmán a, C. Cruz-Hernández a,*, R.M. López-Gutiérrez, E.E. García-Guerrero, *Synchronization of Chua's circuits with multi-scroll attractors: Application to communication,* Commun Nonlinear Sci Numer Simulat, 14 (2009) pp. 2765–2775.

Inzunza González E. was born in Navolato, Sinaloa México in 1976. Received the Bachelors degree in Electronic Engineer from the Culiacán Institute of technology, in 1999, the M. Sc. degree in electronics and telecommunications from CICESE, México, in 2001. Since August 2008, he has been a PhD student. He has 9 years working at the Baja California Autonomous University (UABC) in Ensenada, Baja California, México. His current research interest include synchronization of complex dynamical systems, and cryptography using chaos of biometrics information.

Cruz Hernández C. received the M.S. and Ph.D. degrees in electrical engineering from CINVESTAV, México, in 1991 and 1995, respectively. Since 1995, he is with the Department of Electronics and Telecommunications of the Scientific Research and Advanced Studies of Ensenada (CICESE), where, he is current Professor of Automatic Control. His research interests include multimode oscillations of coupled oscillators, nonlinear systems analysis, and synchronization and control of complex dynamical systems.

López-Gutiérrez R. M. was born in 1972. She is a Professor of Electronics Engineering in Baja California Autonomous University since 2001. She received her Master Science degree and Ph.D. degree in Electronics and Telecommunications from CICESE, Mexico in 1996 and 2003, respectively. Her research interests involve synchronization of complex systems and applications.

Garcia-Guerrero E. E. studied physics engineering at the University Autonomous Metropolitana, Mexico, and received the PhD degree in optical physics from the Scientific Research and Advanced Studies Center of Ensenada, B.C, (CICESE) Mexico. He has been with the Engineering Faculty, Baja California Autonomous University (UABC) Mexico since 2004. His current interests are in the field of Optical Synchronization of Complex Systems.

Cardoza-Avendaño L. was born in Ensenada, B.C. México in 1980. She is Professor of Electronics Engineering in Baja California Autonomous University since 2005. She received her Master Engineering degree in Electrical Engineering from Baja California Autonomous University, México, in 2008. Since August 2008, she has been a PhD student. Her research interests involve synchronization of complex systems and Applications.

Serrano Guerrero H. was born in Culiacán, Sinaloa, México. Received the Bachelors degree in electronic engineer from the Culiacán Institute of technology in 2000, the M. Sc. degree in instrumentation and control from CICESE, México, in 2002. From March 2003 to August 2006 he worked as Test Engineer in Electrónica Lowrance de México. From August 2006 to August 2008 he was a full time professor in the Faculty of Engineering of the Universidad Autónoma de Baja California, México. Since August 2008, he has been a doctoral student and subject professor in the Faculty of Engineering of the Universidad Autónoma de Baja California, México. His current research interest includes control and synchronization of complex dynamical systems, and encrypted transmissions using chaos.