Public-Key hybrid cryptosystem based on Blowfish and RSA

Elza Jintcharadze
Faculty of Informatics and Control Systems
Georgian Technical University
Tbilisi, Georgia
elza.jincharadze@gmail.com

Maksim Iavich
Cyber Security Department
Caucasus University,
Tbilisi, Georgia
m.iavich@scsa.ge

***Abstract*— Nowadays data security is one of the important issues, especially for increasing transactions via the internet. This paper presents a hybrid cryptosystem using RSA (Asymmetric) and Blowfish (Symmetric) algorithm. Hybrid encryption is a combination of symmetric and asymmetric encryption methods. Symmetric algorithms are mostly used for encryption of messages than asymmetric.**

The objective of this research is to evaluate the performance of RSA, Blowfish cryptography algorithms and RSA&Blowfish hybrid cryptography algorithm. The performance of the implemented encryption algorithms is evaluated by means of encryption and decryption time and memory usage. To make comparison experiments, for those algorithms is created program implementation. The programming language Java is used for implementing the encryption algorithms.

Keywords— Symmetric cryptography, Asymmetric cryptography, Data encryption, Ciphertext, Decryption, Hybrid cryptosystem.

I. INTRODUCTION

Nowadays strength of the cryptosystem cannot be totally ensured. The main goal of all cryptography algorithms is to offer the best security, but due to the fact that technology is rapidly developing proposed security systems becoming less resistant to every known or new attacks.

Both symmetric and asymmetric key algorithms have their advantages and disadvantages. Symmetric key algorithms are faster than asymmetric algorithms. The main requirement is that the secret key must be shared in a secure way. Asymmetric systems provide secure transmission of keys, but this process needs much more time. To improve this problem is used the hybrid algorithm, which means using a different type of cryptosystems together [22].

II. RSA

RSA is founded in 1977 is a public key cryptosystem. RSA is an asymmetric cryptographic algorithm named after its founders Rivest, Shamir &Adelman [5]. In general, RSA cryptosystem is used to provide privacy and ensure the authenticity of digital data. Nowadays RSA is implemented in many commercial systems. RSA is used to ensure privacy and authenticity for web servers and browsers, to provide security for web Email and remote login sessions for credit-card payment systems. RSA is frequently used in applications where the security of digital data is important.

RSA generates two keys: a public key for encryption and private key to decrypt the message. RSA algorithm can be divided into three steps: the first step is to generate a key which can be used as key to encrypt and decrypt data; Second step is encryption, where plaintext is converted into ciphertext; and the third step is decryption, where encrypted text is converted in to plain text at another side. RSA is based on the factoring problem of finding the product of two large prime numbers. Key size is 1024 to 4096 bits [5].

The negative side of RSA algorithm is the low speed of encryption. Because the encryption and decryption process with RSA algorithm needs more time than other algorithms. As other symmetric encryption systems, RSA uses two different keys: A public and a private one. Both keys work corresponding to each other, which means that a message encrypted with one of them can only be decrypted by its counterpart. The latter is usually available to the public because private key cannot be calculated from the public key.

III. BLOWFISH

Blowfish is one of the symmetric key algorithms with a 64-bit block cipher and it was developed by Bruce Schneider [1]. Blowfish is a block cipher, the encryption process, and the decryption, Blowfish divides a message into blocks of equal size in length, i.e. 64 bits. Nowadays blowfish provides good security level and there is no successful crypto attack against it. By encryption time Blowfish is faster than DES, but the weak point for this algorithm is its weak key.

IV. DESCRIPTION OF HYBRID CRYPTOSYSTEM

Hybrid encryption is a method of encryption that combines two or more encryption systems. It integrates a combination of asymmetric and symmetric encryption to take benefit from the strengths of each form of encryption. These strengths of the algorithm are defined as speed and security of this algorithm. Hybrid encryption is considered a highly secure type of encryption as long as the public and private keys are fully secure [4].

A hybrid encryption scheme is one that combines the convenience of an asymmetric encryption scheme with the effectiveness of the symmetric encryption scheme. There are various advantages of the combination of encryption methods. One is that users have the ability to communicate through hybrid encryption. Usually, during the encryption process, the asymmetric algorithm is slowing down the encryption process [15]. But hybrid cryptosystem is using

symmetric encryption synchronously so both forms of encryption (symmetric and asymmetric) are improved. The result of the hybrid encryption process has an additional security level with overall improved system performance.

Symmetric and asymmetric cryptography algorithms have their own advantages and disadvantages. In general, symmetric ciphers are considerably faster than asymmetric ciphers, but require all parties to somehow share a secret key. Also, we have to take into consideration that asymmetric algorithms allow public key arrangements and key exchange systems, but this slowdowns encryption process speed [4]. A hybrid cryptosystem is using multiple ciphers of different types together, each to its best advantage. One common method of a hybrid cryptosystem is to generate a random secret key for a symmetric cipher, and then encrypt this key via an asymmetric cipher using the recipient's public key. After this step, the plaintext is encrypted using the symmetric cipher and the secret key. After the encryption process, the encrypted secret key and the encrypted message will be then sent to the receiver.

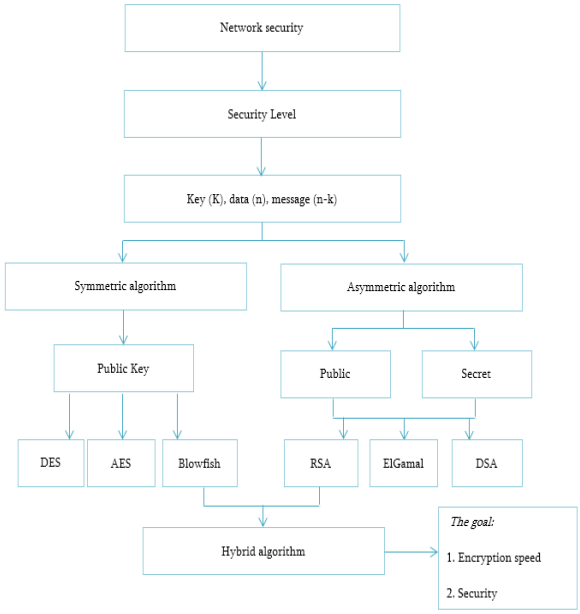

Fig. 1. Main idea of proposed hybrid cryptosystem

The main goal of a hybrid cryptosystem is to generate a random key for the symmetric system and after this encrypt this key for the asymmetric system. So, we will get a secret key that will be used for encryption plaintext. During Hybrid encryption process data are transferred using unique session keys along with symmetrical encryption. The public key encryption process is implemented for random symmetric key encryption. After the receiver gets the encrypted message, the public key encryption method is used to decrypt the symmetric key. After recovering of the symmetric key, then it is used to decrypt the message.

V. Proposed Work – Hybrid Cryptosystem with Combination of Blowfish and RSA

To create a strong encryption algorithm there is a proposed combination of two encryption algorithms - Blowfish and RSA. There were done experiments on proposed algorithms by terms of their encryption speed, used memory and system requirements. The programming language Java is used for implementing the encryption algorithms. To make more exact calculations was used console work with Java NetBeans IDE.

In general encryption time is connected to algorithm architecture. Table 1 shows encryption and decryption results on Blowfish algorithms. Size of used key is 16 bits.

TABLE I. Statistical results of Blowfish Encryption and Decryption process

Plain text size (KB)	Plaintext size (Bytes)	Blowfish Encryption Time (Nanoseconds)	Blowfish Decryption time (Nanoseconds)	Blowfish Encrypted File size (KB)	Blowfish used RAM (Bytes)
32	32710	10753053	1984528	59241	9762104
64	65420	12169867	2743007	119493	10696784
128	130840	12567266	5602025	236670	12556416
256	261680	18200673	9356337	475738	16252696
512	523360	23987822	16802548	954280	23511600
1024	1048460	35550482	26062972	1915678	15407800
2048	2096920	43489299	40463494	3804367	28875368
4096	4193840	62097598	56950097	7552059	55642240

The same experiment was done on RSA system, where was used different size of plaintext. Table 2 shows used encryption time in nanoseconds.

TABLE II. Statistical results of RSA Encryption and Decryption process

Plaintext size (KB)	Plaintext size (Bytes)	RSA Encryption time (nanoseconds)	RSA Decrypted file size (KB)	RSA Decryption Time (Nanoseconds)	RSA Used RAM (Bytes)
32	32710	1536637771	118780	55542452	5611360
64	65420	3208498484	237689	121344997	4677800
128	130840	6149709140	474654	284935252	62035768
256	261680	10574937240	946614	671696785	72146728
512	523360	20368096461	1896331	1991097468	117161952
1024	1048460	41504791208	3795983	6934459468	238824584
2048	2096920	89946149790	7586016	27974097086	371242008
4096	4193840	181620236481	15179673	121238321204	572478144

The proposed hybrid cryptosystem works as following at first system reads plaintext and generates the secret key with RSA and public keys are generated automatically. The next step is to generate Blowfish symmetric key which will be encrypted with RSA system. This provides high security for the key because the usage of RSA algorithm decreases the decryption probability of public key. So, when we share the public key, will be shared also RSA secret key. After these steps, the plaintext is encrypted using Blowfish, because as other symmetric algorithms Blowfish is fast. The decryption process is the reverse process of the above-

described encryption.

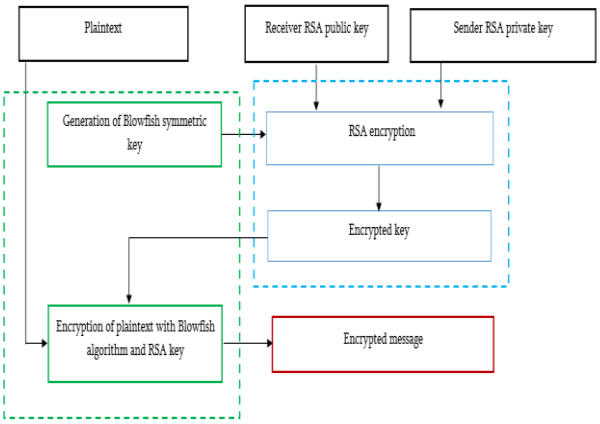

Fig. 2. RSA + Blowfish - the proposed hybrid system architecture

There was created program, implementation for this hybrid cryptosystem on Java programming. Table 3 shows program execution results on different size plaintext.

TABLE III. BLOWFISH + RSA HYBRID SYSTEM ENCRYPTION TIME

Plaintext size (KB)	Plaintext size (Bytes)	RSA+Blowfish Encryption time (nanoseconds)	RSA + Blowfish Encrypted File size (KB)	RSA + Blowfish Decryption time (nanoseconds)	RSA + Blowfish Used RAM (Bytes)
32	32710	9047797	59355	1881211	9498968
64	65420	12203366	118428	2189046	22598000
128	130840	13555651	237417	5057937	26353056
256	261680	14240434	477370	9345405	27380576
512	523360	29886045	951418	18116046	29011368
1024	1048460	40855251	1898922	25666278	30336472
2048	2096920	43979084	3813804	44415486	43218240
4096	4193840	63542269	7624638	54848853	56610432

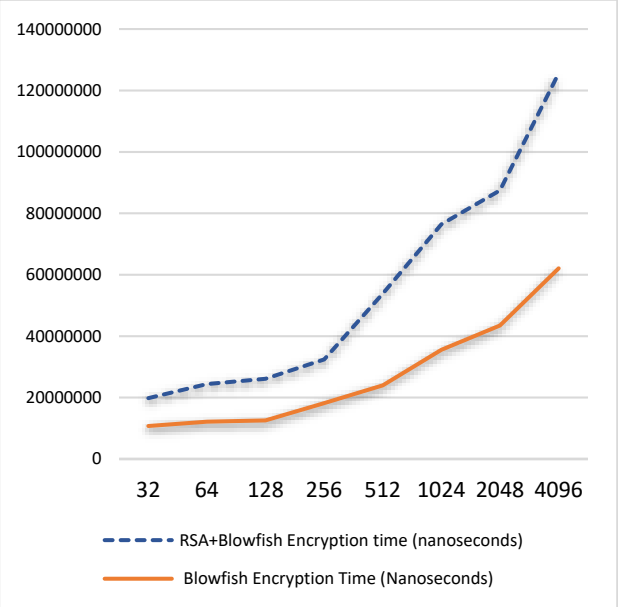

Fig. 3. Comparison of Blowfish and RSA + Blowfish cryptosystems encryption time

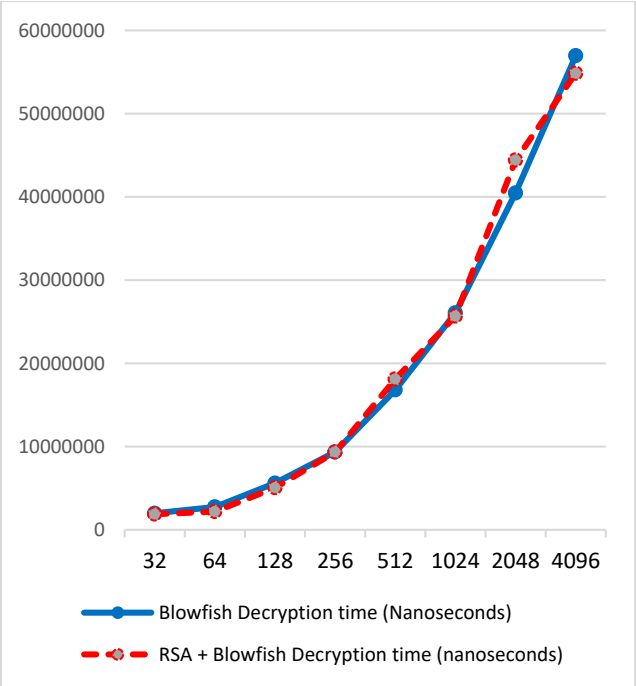

Fig. 4. Comparison of Blowfish and RSA + Blowfish cryptosystems decryption time

VI. CONCLUSION AND SCOPE OF FUTURE WORK

This paper provides a description and comparative analyses of the new hybrid cryptosystem model. The new hybrid model combines Blowfish (symmetric) and RSA (Asymmetric) cryptosystems. The paper shows program implementation and experimental research results with java programing language. Described algorithms and hybrid models are evaluated by terms of encryption speed, memory usage, encrypted file size and ensured security level. Taking into account the time and consumption of the technical resources, Blowfish is the best one else than the other reviewed.

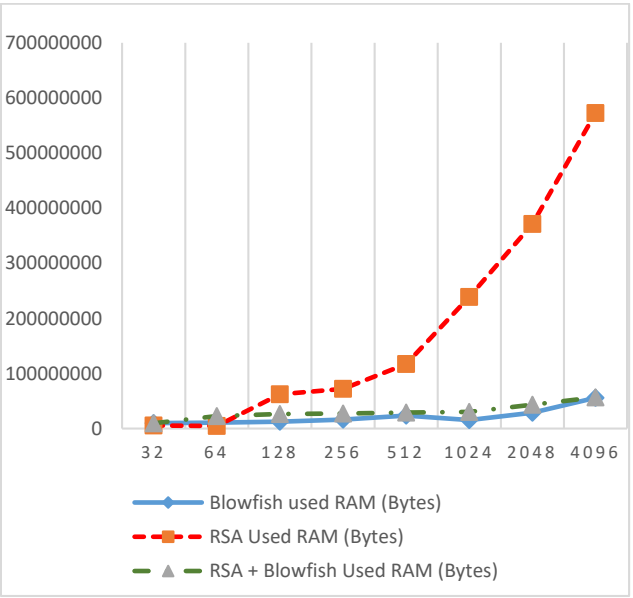

Fig. 5. Used memory comparison chart - Blowfish, RSA and Blowfish+RSA

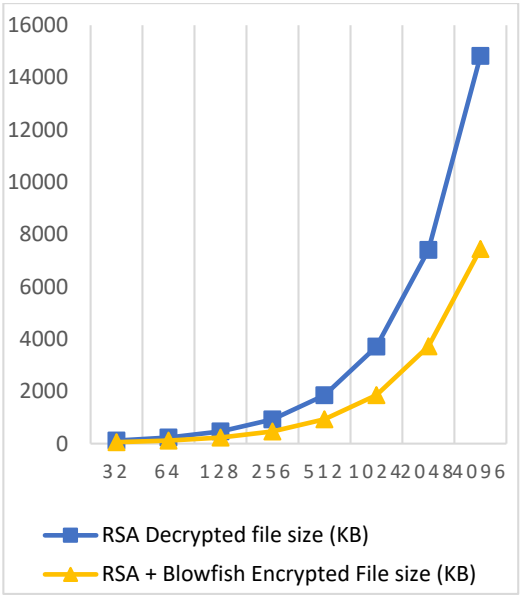

Fig. 6. Encrypted file size comparison chart - RSA and Blowfish+RSA

As a conducted experimental result shows provided new hybrid model is significantly faster and secure because it takes all advantages and strength of symmetric and asymmetric systems. The experiment showed the following results:

- If Blowfish, RSA and Blowfish + RSA hybrid algorithms are compared according to the memory used, the highest technical resources require RSA algorithm, and Blowfish is slightly behind the Blowfish + RSA hybrid scheme.
- Considering the option of encryption Blowfish keeps its initial first position and is the fastest of these systems. However, the Blowfish + RSA hybrid algorithm is far below and significantly faster than RSA. And RSA takes the longest time to encrypt and is very slow.
- Observation of the decryption time parameters has shown that the Blowfish + RSA hybrid algorithm and the blowfish algorithm are almost equally fast with the decryption process and are faster than the RSA algorithm.
- As an overview of the encrypted file size setting, the lowest memory needs Blowfish system, the following is Blowfish + RSA, and the RSA algorithm increases the size of an encrypted file with the highest rate.

For the future is possible to review another hybrid model of symmetric and asymmetric algorithms. It is possible to conduct a series of entropy research of the different cryptographic algorithms and the above-presented hybrid model. This will allow us to identify the sustainability of each algorithm against different types of attacks, including the frequency analysis of the encrypted text.

REFERENCES

[1] B. Schneier, “Description o f a New Variable-Length Key,64-Bit Block Cipher (Blowfish)”, Fast Software Encryption, Cambridge Security Workshop proceedings (December 1993), Springer-Verlag, 1994, pp. 191-204

[2] "The Digital Millennium Copyright Act of 1998" (PDF). United States Copyright Office. Retrieved 26 March 2015.

[3] Cramer, Ronald; Shoup, Victor (2004). "Design and Analysis of Practical Public-Key Encryption Schemes Secure against Adaptive Chosen Ciphertext Attack"

[4] Hofheinz, Dennis; Kiltz, Eike (2007). "Secure Hybrid Encryption from Weakened Key Encapsulation". Advances in Cryptology - CRYPTO 2007

[5] Johhanes A. Buhman, Introduction to Cryptography, Second Edition, 2000

[6] Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanston, Handbook of Applied Cryptography, Massachusetts Institute of Technology, June 1996

[7] Ilya KIZHVATOV, Physical Security of Cryptographic Algorithm Implementations, L’UNIVERSITÉ DU LUXEMBOURG, 2009

[8] Simson Garfinkel, Alan Schwartz, Gene Spafford, Practical UNIX and Internet Security, 3rd Edition Securing Solaris, Mac OS X, Linux & Free BSD

[9] The official Advanced Encryption Standard". Computer Security Resource Center. National Institute of Standards and Technology. Retrieved 26 March 2015.

[10] Баричев С. В. Криптография без секретов. – М.: Наука, 1998.

[11] Шнайер Б. Прикладная криптография. Протоколы, алгоритмы и исходные тексты на языке С, 2-е изд . – М.: Вильямс, 2003.

[12] "Quantum cryptography: An emerging technology in network security". - Sharbaf, M.S. IEEE International Conference on Technologies for Homeland Security. 2011

[13] Adleman, Leonard M.; Rothemund, Paul W.K.; Roweis, Sam; Winfree, Erik (June 10–12, 1996). On Applying Molecular Computation To The Data Encryption Standard. Proceedings of the Second Annual Meeting on DNA Based Computers. Princeton University.

[14] Cramer, Ronald; Shoup, Victor (2004). "Design and Analysis of Practical Public-Key Encryption Schemes Secure against Adaptive Chosen Ciphertext Attack"

[15] Hofheinz, Dennis; Kiltz, Eike (2007). "Secure Hybrid Encryption from Weakened Key Encapsulation"

[16] Taher ElGamal (1985). «A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms

[17] Ященко В. В. Введение в криптографию. СПб.: Питер, 2001.

[18] Hamdan O. Alanazi, B. B. Zaidan, A. A. Zaidan, Hamid A. Jalab, M. Shabbir and Y. Al-Nabhani, “New Comparative Study Between DES, 3DES and AES within Nine factors”, Journal of Computing, Volume, 2, Issue 3, March 2010, pp. 152-157.

[19] Dr. Prerna Mahajan and Abhishek Sachdeva, “ A study of Encryption Algorithms AES, DES and RSA for Security”, Global Journal of Computer Science and Technology Network, Web & Security, Volume 13 Issue 15 Version 1.0 Year 2013, pp. 15-22.

[20] Deepak Kumar Dakate and Pawan Dubey, “Performance comparison of Symmetric Data Encryption Techniques”, International Journal of Advanced Research in Computer Engineering and Technology, Volume 3, No. 8, August 2012, pp. 163-166.

[21] Sumitra, “Comparative Analysis of AES and DES security Algorithms”, International Journal of Scientific and Research Publications, Volume 3, Issue 1, January 2013, pp. 1-5.

[22] Maksim Iavich ; Sergiy Gnatyuk ; Elza Jintcharadze ; Yuliia Polishchuk ; Roman Odarchenko, “Hybrid Encryption Model of AES and ElGamal Cryptosystems for Flight Control Systems”, Oct. 2018

[23] Damaševičius, R., Napoli, C., Sidekerskienė, T., & Woźniak, M. (2017). IMF mode demixing in EMD for jitter analysis. Journal of Computational Science, 22, 240-252.