International Journal of

Electronic Security and Digital Forensics

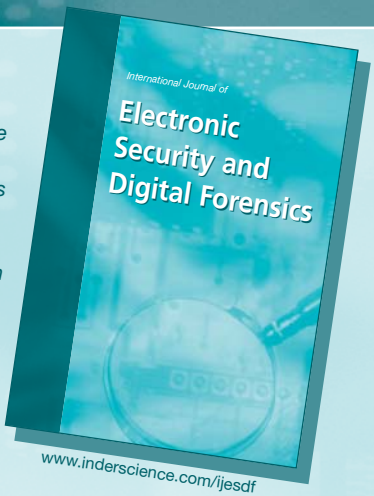
Editor-in-Chief:
Prof. Hamid Jahankhani

Visit www.inderscience.com/ijesdf
for more information and sample articles

INDERSCIENCE
PUBLISHERS
www.inderscience.com

Scope of the Journal

ISSN: 1751-911X (Print), ISSN: 1751-9128 (Online)



IJESDF aims to establish dialogue in an ideal and unique setting for researchers and practitioners to have a knowledge resource, report and publish scholarly articles and engage in debate on various security-related issues, new developments and latest proven methodologies in the field of electronic security and digital forensics. This includes the measures governments must take to protect the security of information on the Internet, the implications of cyber-crime in large corporations and for individuals, vulnerability research, zero day attacks, digital forensic investigation, ethical hacking, anti-forensics, identity fraud, phishing, pharming, relevant case studies, and "best practice" for tackling cyber crime.

Topics covered include:

- Electronic security, information security systems, systems and network security
- Vulnerability research, ethical hacking, zero day attack, attack pattern recognition
- Computational immunology, authentication authorisations
- Security in mobile platforms, mobile agents/artificial intelligence
- Security: security policies/procedures, strategic approaches, requirements engineering
- Identity: theft, management systems, access management systems
- Open source intelligence, criminal data mining/network analysis/intelligence
- Phishing/pharming/spearphishing, cyber war, cybercrime detection/analysis
- Digital cities, GSM-solicited crime
- Computer/mobile device/network/software forensics, anti-forensics
- Digital forensics tools/techniques/standardisation, testing/approvals for forensic tools
- Crime scene/search and seizure processes, criminal investigation of mobile devices
- Investigative techniques, judicial processes, legal/ethical issues, cyber crime legislations
- Digital and physical surveillance, digital image manipulation
- Cryptographic algorithms/protocols, steganography, hidden data

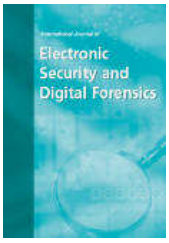
Not sure if this title is the one for you?

Visit the journal homepage at **www.inderscience.com/ijesdf** where you can:

- View sample articles in full text HTML or PDF format
- Sign up for our free table of contents new issue alerts via e-mail or RSS
- View editorial board details
- Find out about how to submit your papers
- Find out about subscription options, in print, online or as part of a journals collection

You can order online at **www.inderscienceonline.com** or download an order form from **www.inderscience.com/subform.**

This title is part of the Computing and Mathematics Collection (see **www.inderscience.com/cm**). For library collection subscriptions or for a free institutional online trial, please contact **subs@inderscience.com.**



International Journal of Electronic Security and Digital Forensics

2021 Vol.13 No.6

Pages	Title and author(s)
571-599	**Low complexity cybersecurity architecture for the development of ITS in smart cities** Nawal Alsaffar; Wael M. El-Medany; Hayat Ali **DOI**: 10.1504/IJESDF.2021.118544
600-611	**Network forensics investigation: behaviour analysis of distinct operating systems to detect and identify the host in IPv6 network** Abdullah Ayub Khan; Syed Asif Ali **DOI**: 10.1504/IJESDF.2021.118542
612-629	**Improving the asymmetric encryption algorithm based on genetic algorithm, application in online information transmission** Le Dinh Son; Tran Van An; Nguyen Ngoc Thuy **DOI**: 10.1504/IJESDF.2021.118543
630-651	**Network and hypervisor-based attacks in cloud computing environments** Reza Montasari; Stuart Macdonald; Amin Hosseinian-Far; Fiona Carroll; Alireza Daneshkhah **DOI**: 10.1504/IJESDF.2021.118549
652-670	**Digital watermarking of compressed videos using larger dimension 2D error correcting codes for higher embedding capacity** Anjana Rodrigues; Archana Bhise **DOI**: 10.1504/IJESDF.2021.118546

Sign up for new issue alerts
Subscribe/buy articles/issues
View sample articles
Latest issue contents as RSS feed
Forthcoming articles
Journal information in easy print format (PDF)

Publishing with Inderscience: ethical guidelines (PDF)
Recommend to a librarian (PDF)
Feedback to Editor

Find related journals

Keep up-to-date
Our Blog
Follow us on Twitter
Visit us on Facebook
Our Newsletter (subscribe for free)
RSS Feeds

671-684 **Volatile memory forensics of privacy-aware operating systems**
Nilay R. Mistry; Sampada Kanitkar; S.O. Junare
DOI: 10.1504/IJESDF.2021.118548

Improving the asymmetric encryption algorithm based on genetic algorithm, application in online information transmission

Le Dinh Son and Tran Van An

Le Quy Don Technical University,
236 Hoang Quoc Viet, Hanoi 100000, Vietnam
Email: sonld@lqdtu.edu.vn
Email: tavistu@gmail.com

Nguyen Ngoc Thuy*

Vietnam National Mine Action Centre,
Thach Hoa Commune, Thach That District,
Hanoi, 100000, Vietnam
Email: nnthuy.vnmac@gmail.com
*Corresponding author

Abstract: Within the paper scope, the authors propose to improve two solutions of information security: First, improving the asymmetric key encryption based on genetic algorithm (GA); second, building architecture of stratified information transmission system with intermediate information transmission layer. The method of survey and analysis is applied with scientific publications related to asymmetric encryption and genetic algorithms. Applying genetic algorithm to improve asymmetric encryption algorithm and intermediate information transmission layer used in the building of information transmission system in order to further enhance the security. Empirical evaluation of the effectiveness of the proposed solutions. Application of proposed solutions in actual system in use. Improved asymmetric encryption algorithm based on genetic algorithm; applied the above algorithm in building a stratified information transmission system with intermediate information layer. The improvement of information security solutions has further reinforced the security and ensured the processing speed as well as prospectively applied in practice.

Keywords: genetic algorithm; information security; asymmetric encryption; information transmission.

Reference to this paper should be made as follows: Son, L.D., An, T.V. and Thuy, N.N. (2021) 'Improving the asymmetric encryption algorithm based on genetic algorithm, application in online information transmission', *Int. J. Electronic Security and Digital Forensics*, Vol. 13, No. 6, pp.612–629.

Biographical notes: Le Dinh Son is a Lecturer in the Faculty of Information Technology of Le Quy Don Technical University. He received his PhD from St. Petersburg Electrotechnical University (LETI). He has 20 years of teaching and researching in various fields of information technology, having published over 20 domestic and international papers. His main research directions include: mathematical assurance for computational systems; optimisation theory and applications; data mining; distributed systems; natural language processing (NLP); and data mining in information security.

Tran Van An is a Lecturer in the Faculty of Information Technology of Le Quy Don Technical University. He received his PhD from Irkutsk State Technical University (IrGTU). He has eight years teaching and researching in various fields. His main research directions include: natural language processing (NLP); genetic programming; and data mining in information security. He has published 15 research papers in journals and seminars.

Nguyen Ngoc Thuy is a researcher at the Vietnam National Mine Action Center. He received his Master's degree from Le Quy Don Technical University, and has seven years of research experience at the National Mine Database Center. His main research directions are data mining, and big data in post-war mine action in Vietnam.

1 Introduction

Information transmission in internet nowadays is witnessed to be conducted by various models such as peer-to-peer, client/server, and hybrid. For better illustration, the transmission model being applied by the authors is client/server, which is relatively common in computer network and contains two main components. The general information transmission architecture of the client/server model is depicted in Figure 1.

Figure 1 Diagram of system components in the client/server model (see online version for colours)

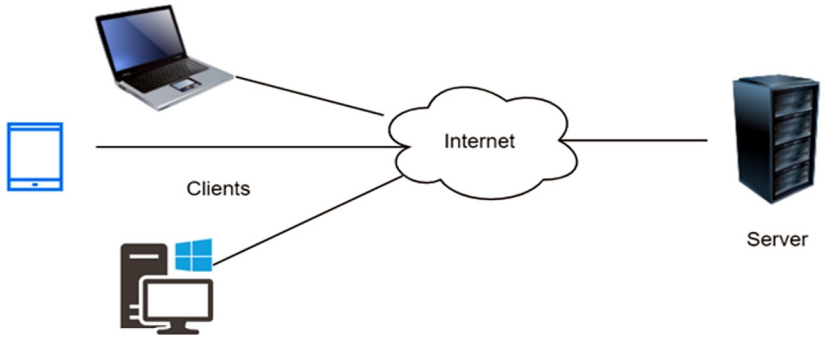


Figure 1 illustrates that the client/server model is deployed in many information systems in a distributed form. Due to the need to exchange information between different computers in remote geographical areas, information security is a major challenge of this model. Those are: security authentication, authorisation, access control and encryption (Karabey and Akman, 2016; Oluwatosin, 2014). There also exists professional risk attached in client/server model namely:

- Users can automate the authentication procedure.
- Clients can be installed in public areas or in a high-risk area.
- Clients can activate utilities or specialised equipment to bypass security mechanisms.
- In extreme cases, users can tamper with and invade the system.

Research (Rao and Selvamani, 2015) conducted surveys on information security challenges in cloud computing platforms (using the client/server model), then published reports on the impact of data loss or data leakage on an organisation's business, brand and trust (Table 1).

Table 1 Information security challenges

Security challenges	*Critical (%)*	*Very important (%)*	*Important (%)*	*Less important (%)*	*Not important (%)*
Data security and privacy	70	25	2	3	0
Compliance issues	30	50	16	4	0
Legal and contractual issues	39	37	21	4	0
Challenges in migration	11	22	38	18	11
Lack of clarity in pay per use model	13	15	47	24	2
Integration of cloud based applications with legacy systems	22	33	33	8	4

As noted in Table 1, the challenges of security and privacy in the client/server model is tremendous. With such risks and challenges, it is crucial to strengthen information security in client/server model. Information security means protecting information and information system from unauthorised access, use, disclosure, damage, modification, and illicit record. System security must be ensured over six criteria: confidentiality, possession, integrity, authenticity, availability and utility (Kumar et al., 2018).

Information security measures are often applied in many levels: hardware, software, infrastructure, etc. In this paper, the authors focus on handling software-level security issues with encrypted information during transmission process. Encryption algorithms are often divided into two types: symmetric and asymmetric encryption. The basic difference between the two encryption methods is that symmetric encryption algorithms use a single key, while the other uses two separated but relevant keys.

With the use of symmetric encryption, the scientific publication (Shaktawat et al., 2020) uses a crossover method between the advanced encryption standard (AES) algorithm and the separation and permutation of data blocks for image encryption (a hybrid approach for image encryption is proposed by combining AES, a standard cryptography algorithm, along with splitting and block permutation). In it, a standard image block is used as an input to enhance security, the image is divided into 4 * 4 matrix, followed by block permutation before using AES for image encryrtion. With this method, the security is significantly enhanced. However, due to the use of AES algorithm, the private key must be stored to avoid being lost. In the transmission of information, key security must be conducted online.

In the scientific publication (Baykara et al., 2017), the author proposes a new symmetric encryption algorithm (novel Symmetric Encryption Algorithm) with the use of algebraic transformations, bit sequence transformations, and mixing arrays of bits. The authors installed and applied algorithms in encoding text files. Nevertheless, this algorithm remains simple with random factors are not optimally utilised. The algorithm in use is the symmetric encryption algorithm, so it is mandatory to hold a security mechanism with a private key if the goal is applying in online information transmission.

The scientific publication (Sivakumar et al., 2017) proposes an advanced secure data encryption algorithm (ES-DES) that uses an extended substitution box (S-Box) to support and improve the security of the DES algorithm. The increase of the key size coupled with S-Box creation and extension helps improve the security of the ES-DES algorithm against attacks. The authors also pointed out the limitation of the algorithm is that the processing speed stays slow, hence, it requires computers with equivalent calculating capacity.

Regarding asymmetric cryptographic algorithms, the scientific publication (Das et al., 2020) has improved the asymmetric encryption algorithm RSA (new modified version of standard RSA), the improvement is made by the substitution of algorithm seeking the value of the public and the private key and proposal of an algorithm to find ciphertext and plaintext in encryption and decryption. These formulas and algorithms alleviate the complex calculations of finding private keys, public keys, ciphertext and plaintext. However, in this scientific publication, the assessment of security after improving the RSA algorithm is unavailable. The application of this algorithm to online information transmission has not been tested, the authors also proposed to design an algorithm for the process of encoding and decoding online information in the coming time.

In association with the use of asymmetric encryption, the authors of the scientific publication (Fang et al., 2019) improve the ElGamal algorithm for application in digital signatures with upgraded security and performance. In terms of security, random elements are increased and the Hash function is used when the public key is generated. In terms of performance, the inverse modular operation is reduced, which increases the execution speed. The authors have analysed the security and performance of the proposed algorithm. However, in terms of security, the author has not provided specific comparison with other algorithms. In terms of performance, the given algorithm is better than ElGamal, however, its use in online information transmission has not been put under analysis.

Upon the basis of aforementioned publications, the disadvantage of symmetric key encryption derives from the requirement for the transmission of shared keys for information encryption and decryption. If these keys are shared by insecure connections, the risk of being interfered by a third party is enormous. When a user is not authorised to gain access to a symmetric key, all information encrypted with that key will be compromised. This drawback is tackled by the asymmetric encryption algorithm by generating a key pair consisting of a public key and a private one, in which the public key is freely distributed and the private key only be kept secret for the owner of the key pair. There go many asymmetric encryption algorithms such as RSA and Elgamal or other algorithms upgraded from this one, but the downside is the slow coding and decoding speed, unease for online information transmission (Verma et al., 2016). Another disadvantage is that, in the case of an attacker purposely modifying the content of code on the transmission line, according to the Elgamal and RSA algorithms, the recipient cannot detect this (Okeyinka, 2015, 2017).

For the above reasons, the authors propose to improve the asymmetric encryption algorithm based on the genetic algorithm (GA) in response to the security of asymmetric encryption combined with the techniques in the GAs such as crossover, mutation, security-increase and invulnerability by complexity with binary transforms, logical operations, XOR to overcome disadvantages of Elgamal and RSA.

Security enhancements are conducted not only at the message-encryption level, but also at the authentication level between the transmitting and receiving parties. There have

been many studies on how to organise the system for authentication during information transmission. In the publications (Ali and Murray, 2016; Dossogne and Lafitte, 2015; Kucharczyk, 2010), the authors proposed an architecture model of stratified system with intermediate authentication layer for online voting. The presence of an authentication intermediate layer enhances the system's security and prevents tampering from transmitters. However, these systems only focus on one-way authentication, meaning that only the transmission is authenticated while the responded message from the receiver is not. In this paper, the authors develop stratified information transmission system architecture with an intermediate authentication layer to enhance information security during transmission in both directions. To boost up the speed of message transmission due to authentication, encryption and decryption of information, the authors applied the methods (Marz and Warren, 2015) with caching information in need of authentication.

2 Methods

2.1 Improve the asymmetric encryption algorithm based on GA (1)

GA refers to the technique imitating the evolutionary adaptation of biological populations based on Darwinism. GA is a method of random optimisation by mimicking the evolution of humans or organisms. The idea of GA is to simulate natural phenomena, to inherit and fight for survival. GA solves mathematical planning problems through basic processes: crossover, mutation and selection for individuals in the population (Nekoei et al., 2015).

Cryptography plays such a vital role in network security. Cryptography is the science of composing in secret code (Sokolov and Shangin, 2002). The purpose of the password is to protect the information transmitted and read by anyone except intended recipients. Ideally, unauthorised individuals will not be able to read encrypted messages.

In cryptography, encryption algorithms generate keys in the form of a bit sequence, which is used to encrypt and decode a piece of information. The way these keys are used makes the difference between symmetric and asymmetric encryption.

The asymmetric cryptography system involves the use of two keys (public and private). The public key may be disclosured, when exchanging messages, the public key is mandatory to be forwarded. An important requirement is ensuring the authenticity of the transmitted public keys (Sokolov and Shangin, 2002). This is interpreted in Asymmetric encryption model depicted in Figure 2.

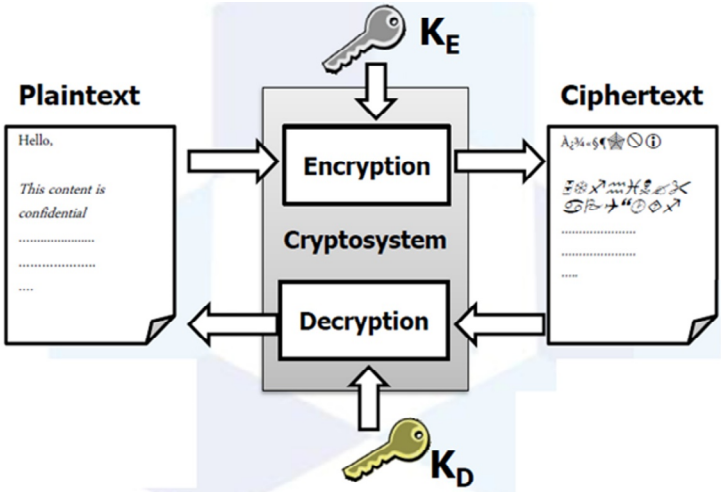
Figure 2 illustrates an asymmetric encryption model, in which:

- Plaintext – original text: The original text is readable and should be protected,
- Ciphertext – encrypted text: Ciphertext is the encrypted output resulting from the application of a cryptographic method on plaintext.
- K_E (K_{public}) – public key used for encryption.
- K_D ($K_{private}$) – private key used for decryption.
- $K_E \# K_D$.

There have been researches on using GA to solve coding problems such as publication (Naik and Naik, 2014), the author exploits the randomness related to crossover and mutation processes to create symmetric key for encrypting and decoding messages. The

algorithm is further enhanced with the use of symmetric key permutations by predefined permutation coefficients as agreed by both sender and receiver. However, the way to create mutations is simple, requiring additional intermediate transformations to strengthen data security. In line with such an approach, publication (Alhussain, 2015) developed a key-exchanging algorithm based on the crossover and mutation features of GAs and asymmetric encryption. The number of crossover points combined with the number of mutant points determines the length of the private key, which increases the confidentiality of the algorithm. The security is enhanced by the transformation of bit blocks; however, the transformation is simple.

Figure 2 Asymmetric encryption model (see online version for colours)



The security of any encryption algorithm is also identified by the encryption key in use. Qualified cryptographic keys must have sufficient length and random bit values. Key distribution is the most important process in key management, it can not only use one or more key distribution centres, but also exchange session keys between the parties.

In this article, the authors improved the key-exchanging algorithm based on a combination of GA and asymmetric encryption, which uses crossover, mutation and especially adds more complex steps to convert blocks of bits into algebraic methods.

A sequence of messages is the object being transmitted between parties, before being transmitted, they are divided and standardised into a set of sub-messages with a length of 16 bytes, of which, 13 bytes are the contents of the sub-message, 1 bytes contain the identifier (id) of the message string, 1 byte is the index number of the sub-message in the sequence, 1 byte contains the checksum of the sub-message in the original message sequence. These three parameters serve to concatenate message blocks after decoding.

Dividing into sub-message sequences ensures higher security, the 1st to through 13th byte contain sub-messages, in case the number of bytes is less than 13, the sub-message will be filled with spaces. The 14th byte carries the message sequence ID in the same generated cycle so that they do not overlap, and receives a value from 0–127, after an ID cycle reaches 127, the ID will be started over from zero. The 15th byte contains the numerical order of sub-messages in the distributed message sequence, the 16th byte encompasses the checksum of sub-messages in the original message sequence, all

16 bytes are encrypted with the public key, sent to the receiver. The receiver collects each message, proceeds with the decoding, and based on the identifier (id), the order (index) and the checksum of sub-messages in order to concatenate the sub-messages into a sequence. The structure of a sub-message sequence is described as follows:

b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9	b_{10}	b_{11}	b_{12}	b_{13}	b_{14-id}	$b_{15-index}$	$b_{16-checksum}$

In this algorithm, the key exchange is based on a combination of GA and asymmetric encryption, combining the cross and mutant points on the basis of complex algebraic transformation. The steps of the process described below are performed on each sub-message (hereinafter referred to as the message).

To conduct encryption, it requires the selection of the size N of the binary block to be split from the original binary sequence, the number of crossover points C, the M mutant points randomly generated in the process. These values will be encrypted and sent on the other side.

Step 1 Convert characters from the message (13 characters) into equivalent values in their ASCII tables.

Step 2 Concatenate the ASCII sequence obtained with id, index and checksum, gain a sequence with 16 blocks.

Step 3 Represent ASCII code in sequence into 8 bits binary strings.

Step 4 Divide the bit sequence into blocks with size N. Thus, the total number of blocks received is $S = 128 / N$.

Step 5 Convert the received block chain into N column matrix, call it matrix A.

Step 6 Randomly generate two numbers n_1 and n_2, $1 \le n_1, n_2 \le N$.

Step 7 Swap two columns n_1 and n_2 in that matrix and the resulting matrix is A_1.

Step 8 Calculate the transposition matrix of the matrix A_1 and gain $B = A_1^T$.

Step 9 Combine the elements of the matrix B into a line in order from left to right, from top to bottom, and then divide the resulting binary line of characters into two equal lines, obtaining two binary lines of 64-bit length. Symbols of ranges that are root1 and root2. These two ranges are represented as follows:

- root1:

r_1	r_2	r_3	r_4	r_5	r_6	⋯	⋯	r_i	⋯	⋯	r_{63}	r_{64}

- root2:

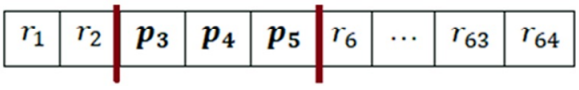
p_1	p_2	p_3	p_4	p_5	p_6	⋯	⋯	p_i	⋯	⋯	p_{63}	p_{64}

In which: $0 \le r_i \le 1$; $0 \le p_i \le 1$; $i = \overline{1, 64}$.

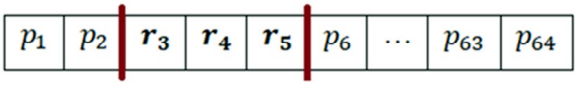
Step 10 Generate the random numbers for the crossover between the two received root1 and root2 ranges, call the number of crossover points C, the random numbers are

c_i, $1 \le c_i \le C$, $i = \overline{1, C}$. Conduct crossover between the two ranges above and the cross points c_i, given c_i carrries values 2 and 5, the result will be two ranges child1 and child2 whose values are as follows:

- Child1: (see online version for colours)



- Child2: (see online version for colours)



Step 11 Randomly generate the number of mutations M and mutant points $_{mi}$, $1 \le m_i \le M$, $i = \overline{1, M}$, and random values range along the length of child1 and child2.

Step 12 Mutations in the child1 and child2 sequences at the mutant points are generated at step 11 with bit NOT operator.

Step 13 Make a left logical shift to child1 and child2 sequences.

Step 14 Divide the child1 and child2 sequences from step 13 into blocks with a length of 8 bits, then combine them into a sequence of bits blocks.

Step 15 Random point generation is used to split blocks of bits, called it a crosspoint. The crosspoint gets values from 1 to 7.

Step 16 Divide blocks of bits into two parts by the crosspoint, then convert those parts into radix 16 and combine the parts into a sequence separated by spaces.

Step 17 The random number consists of an RF random factor from 1 to 15.

Step 18 Establish a sequence of random factor S with $s_i = RF$, $i = \overline{1, length(S)}$, in which: $length(S) = C + M + 6$.

Step 19 Randomly generate a permutation factor (PF) with range from 1 to $(length(S) - 1)$.

Step 20 Establish a private key including: size N of binary blocks, crossover points c_i, mutant points m_j, crosspoints, other points of n_1, n_2 and random coefficient. The components in the block are converted to radix 16; hence, the private key $K_{private}$ holds the following form:

$$N[c_i][m_j]crosspoint\ n_1n_2C\ M\ PF\ RF.$$

Step 21 Conduct permutation of components in a private key segment based on Permutation factor coefficient. The code to be permuted is:

$$N[c_i][m_j]crosspoint\ n_1n_2\ C\ M.$$

The permutation is deployed as follow: PF coefficient is the point to bisect the upper sequence. Making permutations by changing the position of two parts of

the sequence and conducting RF such permutations, the outcome range is so-called $KR_{private}$,

Step 22 Establish a partially public key KF_{public} from $KR_{private}$ under XOR operation:

$$KF_{public} = [KR_{private} XOR\ S].$$

Step 23 The outcome public key is the result of connecting KF_{public} to $PF\ RF$. So, the public key holds a form as:

$$K_{public} = KF_{public}\ \ PF\ \ RF.$$

Step 24 The encrypted message is joined by the character sequence received from step 16 and the public key K_{public}.

The above algorithm is depicted hereinafter, given the input message block is: *&*AcfER5#*.

The size of the binary block $N = 4$, number of crossover points $C = 2$, number of mutant points $M = 4$.

The step results are presented as follows:

- Step1:

  38 42 65 99 102 69 82 53 35 32 32 32 32

- Step 2:

  38 42 65 99 102 69 82 53 35 32 32 32 32 1 1 1

- Step 3:

  00100110 00101010 01000001 01100011 01100110 01000101 01010010
  00110101 00100011 00100000 00100000 00100000 00100000 00000001
  00000001 00000001

- Step 4:

  0010 0110 0010 1010 0100 0001 0110 0011 0110 0110 0100 0101 0101 0010
  0011 0101 0010 0011 0010 0000 0010 0000 0010 0000 0010 0000 0000 0001
  0000 0001 0000 0001

- Step 5:

$$\begin{bmatrix} 0010 & 0110 & 0010 & 1010 \\ 0100 & 0001 & 0110 & 0011 \\ 0110 & 0110 & 0100 & 0101 \\ 0101 & 0010 & 0011 & 0101 \\ 0010 & 0011 & 0010 & 0000 \\ 0010 & 0000 & 0010 & 0000 \\ 0010 & 0000 & 0000 & 0001 \\ 0000 & 0001 & 0000 & 0001 \end{bmatrix}$$

- Step 6:

  n_1 and n_2, given $n_1 = 1, n_2 = 2$

- Step 7: Swap 2 columns n_1 and n_2 in that matrix, the matrix result is A_1

$$\begin{bmatrix} 0110 & 0010 & 0010 & 1010 \\ 0001 & 0100 & 0110 & 0011 \\ 0110 & 0110 & 0100 & 0101 \\ 0010 & 0101 & 0011 & 0101 \\ 0011 & 0010 & 0010 & 0000 \\ 0000 & 0010 & 0010 & 0000 \\ 0000 & 0010 & 0000 & 0001 \\ 0001 & 0000 & 0000 & 0001 \end{bmatrix}$$

- Step 8: Matrix $B = A_1^T$

$$\begin{matrix} 0110 & 0001 & 0110 & 0010 & 0011 & 0000 & 0000 & 0001 \\ 0010 & 0100 & 0110 & 0101 & 0010 & 0010 & 0010 & 0000 \\ 0010 & 0110 & 0100 & 0011 & 0010 & 0010 & 0000 & 0000 \\ 1010 & 0011 & 0101 & 0101 & 0000 & 0000 & 0001 & 0001 \end{matrix}$$

- Step 9: Concatenate the elements of the matrix B into a line in order from left to right, from top to bottom, then divide the resulting binary line of characters into two equal lines.

root1 0110000101100010001100000000000100100100011001010010001000100000

root2 0010011001000011001000100000000010100011010101010000000000010001

- Step 10: Generate random numbers serving crossover:

crossover1 = 15, crossover2 = 23.

child1 0110000101100011001000100000000100100100011001010010001000100000

child2 0010011001000010001100000000000010100011010101010000000000010001

- Step 11: Generate randomly the number of mutant points m_i

$\{12,\ 10,\ 2,\ 17\}$

- Step 12: Mutate two sequence child1 and child2 at mutant points

childMutation1 0100000101001011011000100000000100100100011001010010001000100000

childMutation2 0000011001101010011100000000000010100011010101010000000000010001

- Step 13: Make a left logical shift

childMutation1 left logical shift 1000001010010110110001000000001001001000110010100100010001000000

childMutation2 left logical shift 0000110011010100111000000000000101000110101010100000000000100010

- Step 14:

  10000010 10010110 11000100 00000010 01001000 11001010 01000100 01000000
  00001100 11010100 11100000 00000001 01000110 10101010 00000000 00100010

- Step 15:

  Randomly generate cross – point = 2

- Step 16: Divide the bits block into two parts by the cross-point, converting the parts into radix 16:

  2 2 2 16 3 4 0 2 1 8 3 a 1 4 1 0 0 c 3 14 3 20 0 1 1 6 2 2a 0 0 0 22

- Step 17: Generate random number including random factor $RF = 9$
- Step 18: Establish random number sequence S

  9 9 9 9 9 9 9 9 9 9 9 9

- Step 19:

  Randomly generatepermutation factor $PF = 3$

- Step 20: Establish private key $K_{private}$:

  4 f 17 c a 2 11 2 1 2 2 4

- Step 21: Sequence $KR_{private}$ after RF permutation times:

  c a 2 11 2 1 2 2 4 4 f 17

- Step 22: Establish a partially public key KF_{public}

  5 3 b 18 b 8 b b d d 6 1e

- Step 23: Establish a public key K_{public}

  5 3 b 18 b 8 b b d d 6 1e 3 9

- Step 24: The encrypted message:

  2 2 2 16 3 4 0 2 1 8 3 a 1 4 1 0 0 c 3 14 3 20 0 1 1 6 2 2a 0 0 0 22 5 3 b 18 b 8 b b d d 6 1e 3 9

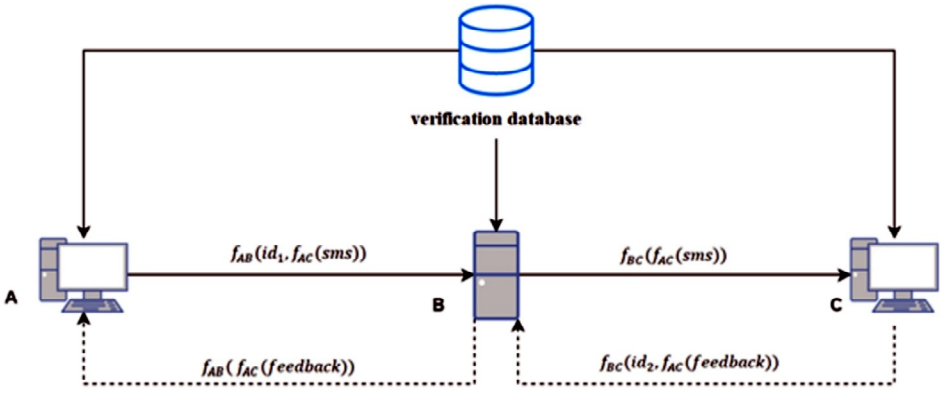
2.2 Stratified information transmission architecture

In addition to encryption in the content of transmitted messages, many studies have suggested strengthening security during information transmission with multiple layers. Publications (Ali and Murray, 2016; Dossogne and Lafitte, 2015; Kucharczyk, 2010) study the technique of forming a security protocol for transmitting information between computers in the system, in which, encrypting different pieces of information in packets, in different transmitting steps, this protocol is entitled blind intermediaries. Based on this idea, the authors apply the aforementioned encryption algorithm and develop additional protocols to further security, in which the intermediate layer participates in all phases of in-and-out transmitted messages. The basic techniques to ensure information confidentiality are: encryption, authentication of parties, splitting of messages,

verification of transmitted information to check integrity, checking real-time information during transmission. Algorithms have been developed to perform a distinct sequence of actions at each stage.

During information transmission, information security is enhanced via an ID verification mechanism, which creates mutual trust between the parties. One of the ways to ensure the authenticity of the parties is to use the 'request-feedback' mechanism. Information integrity will be controlled by checksum parameters in algorithm (1), which is guaranteed throughout the transmission. Information needs transmitting over a certain period of time, are processed synchronously and impossible to reuse in another moment. One of the options to control time-based information transfer is time-based server-side session monitoring. The server accepts the client connection for a while and if it fails, it will disconnect.

Figure 3 Connection diagram in a stratified information transmission architecture (see online version for colours)



Based on this idea, the authors implemented the system architecture as presented in Figure 3. In order to deliver the message from the starting point A to the end point C, the system forms an intermediate point B, where B is responsible for decoupling, validating information transmitted from A. However, B only partially decodes, then transmits the remainder to C. The process of transmitting messages between components is carried out according to the proposed algorithm mentioned above.

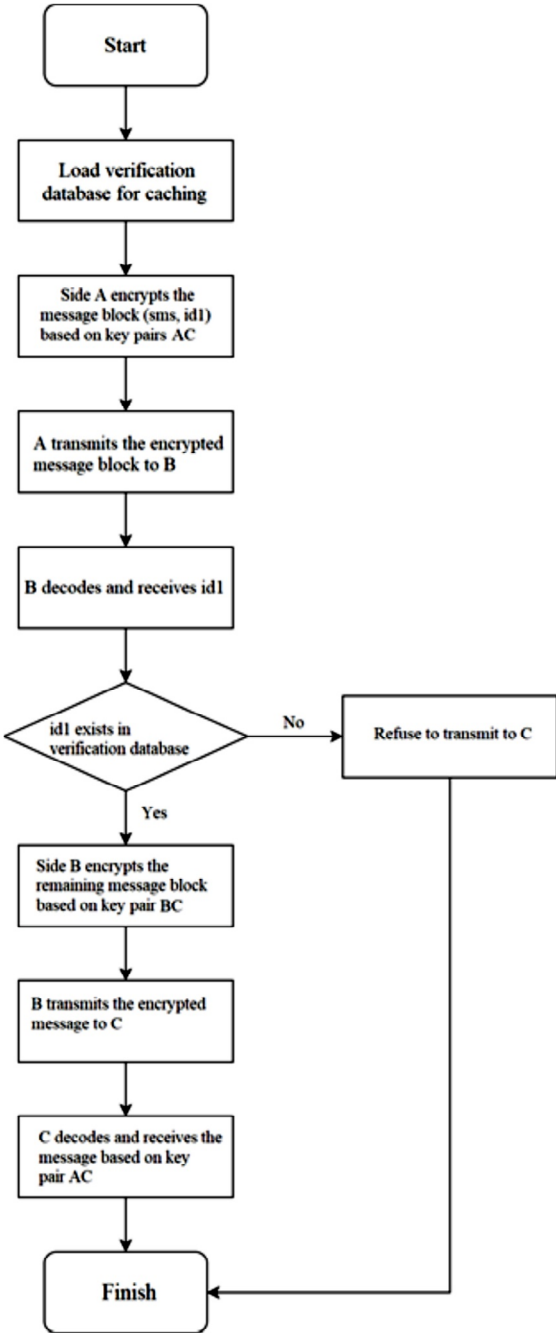
Figure 3 depicts the connection model between parties A and C through B. Based on the private-key communication protocol, key pairs according to sessions AB, BC, AC are generated:

- AB – Session key pair between A and B
- AC – Session key pair between A and C
- BC – Session key pair between B and C.

This means that, each time the message is transmitted, between AB, AC, BC generates $K_{private}$ and K_{public} key pairs.

The algorithm to deliver messages from A to C through B is illustrated in Figure 4.

Figure 4 Algorithm for transmitting messages from A to C via B



During the implementation of the algorithm in Figure 4, to ensure speeding up the querying of identifiers for authentication and real-time response, the authors use data caching solutions. All authentication data is transferred from the database to all three sides A, B, C before transmitting the message. Side A encrypts the SMS message based

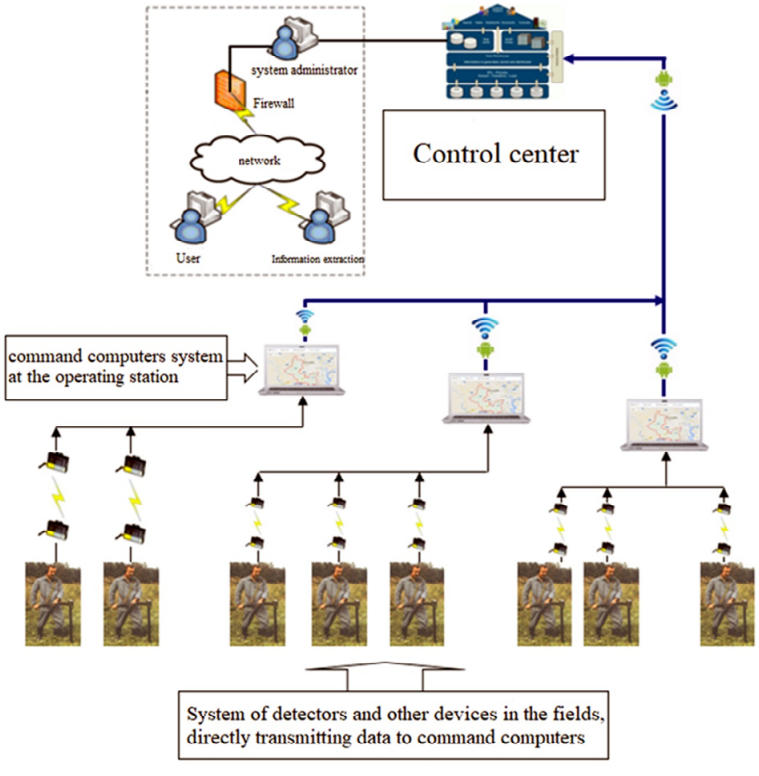
on the AC key, then attaches an identifier id_1 distributed from the authentication database, then continues to encrypt this message block on the AB key and delivers the message. This has been encrypted for side B.

The B side in this case is so-called an intermediate layer as it can only decode part of the message to retrieve id_1 for authentication, and the remaining information cannot be decoded by B side. The B side receives the message, decodes, checks whether the identifier is in the database or not. In case of id_1 existence in the authentication database, B continues to encrypt the remaining message with the BC key and sends it to C. After receiving and decoding sms information, the C side codes the *feedback* and sends to B and continue to A in reverse order. The principle is to ensure that: *SMS* messages will only be accepted if id_1 is present in the authentication database. Based on these principles, an online information transmission system can be built in which the main requirements are met including the information is encrypted and guaranteed not to be tampered with.

2.3 Deployment of information transmission system

The improvement of the asymmetric encryption algorithm upon the GA proposed in the article has well handled the security problem by using many techniques of GA such as crossover, mutations, binary transformations, algebraic transformations, using logical operations.

Figure 5 Information transmission architecture in information management system for mine action in Vietnam (see online version for colours)



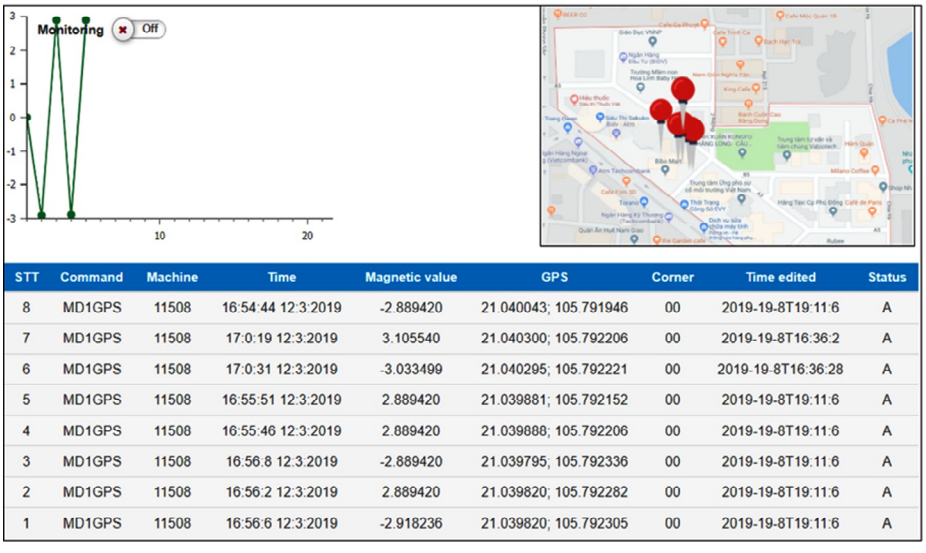
The above algorithm deals with code detection according to probability of occurrence. Symmetric encryption methods share the common drawback of creating the same identical blocks of text with the original ones. Thanks to GA techniques: crossover, mutation, algebraic transformations with random values, creating different charsets with the same original text as well as completely eliminating probabilistic code detection.

In case of an attacker intentionally modifying the content of coded text on the transmission line, supposed with the use of Elgamal and RSA algorithms, the receiver cannot detect this, our proposed algorithm can detect this thanks to the XOR technique.

With the proposed algorithms and models, the authors have applied it to the Information management system for mine action in Vietnam. The communication architecture is described in Figure 5.

In Figure 5, the object information collected by field detectors is transmitted to command computers at the control station via radio signals and is digitised. From the command computers at the control station, the signal is transmitted to the dispatch centre via the internet with the client/server model. The authors apply the security methods studied in the paper during the stage of transmission from the command computers at the control station to the dispatch centre with an intermediate authentication layer. The system has met the real-time transmission. The system has been being put into trial operation at the Vietnam National Mine Action Center (VNMAC) (Figure 6).

Figure 6 Experimental software interface for application (see online version for colours)



STT	Command	Machine	Time	Magnetic value	GPS	Corner	Time edited	Status
8	MD1GPS	11508	16:54:44 12:3:2019	-2.889420	21.040043; 105.791946	00	2019-19-8T19:11:6	A
7	MD1GPS	11508	17:0:19 12:3:2019	3.105540	21.040300; 105.792206	00	2019-19-8T16:36:2	A
6	MD1GPS	11508	17:0:31 12:3:2019	-3.033499	21.040295; 105.792221	00	2019-19-8T16:36:28	A
5	MD1GPS	11508	16:55:51 12:3:2019	2.889420	21.039881; 105.792152	00	2019-19-8T19:11:6	A
4	MD1GPS	11508	16:55:46 12:3:2019	2.889420	21.039888; 105.792206	00	2019-19-8T19:11:6	A
3	MD1GPS	11508	16:56:8 12:3:2019	-2.889420	21.039795; 105.792336	00	2019-19-8T19:11:6	A
2	MD1GPS	11508	16:56:2 12:3:2019	2.889420	21.039820; 105.792282	00	2019-19-8T19:11:6	A
1	MD1GPS	11508	16:56:6 12:3:2019	-2.918236	21.039820; 105.792305	00	2019-19-8T19:11:6	A

In Figure 6, the information is directly presented on the chart, the map as well as in the form of tables, including information about the magnetic signal, time and GPS.

3 Results and discussion

The improvement of the asymmetric encryption algorithm based on GA in collaboration with the building of stratified information transmission system architecture and

intermediate information transmission layer has strengthened information security, simultaneously met the speed of encryption and decryption of computers.

The author has compared the security of the proposed algorithm (1) with the Advance Elgamal algorithm (Dossogne and Lafitte, 2015). Cryptanalysis method is applied to evaluate the security of an algorithm. One of the common approaches is: Brute force.

When applying the Brute force for cryptanalysis on the Advance Elgamal algorithm with a message of 64-byte length, according to calculations of the scientific publication (Pham and Nguyen, 2006), the key space set holds a size of 2^{192}. Provided a computer with a capacity of 20000 billion ($2*10^{13}$) calculations per second, the estimated time for cryptanalysis is $1.6*10^{37}$ years.

With the message segment size of 16 bytes (according to the theory, the complexity of detecting the code is less than 64 bytes), to successfully detect the message encoded by the proposed algorithm (1), in addition to overcoming the difficulty of algebraic transformations, it is mandatory to find the exact sequence of bits in steps 4, 10 and 12. The values of these bits are the result of transformations through random parameters generated during the process of crossover and mutation; thus, the key space enlarged enormously. The size of the key space set is:

$$SV = 2^{128} * 2^{64} * 2^{64} = 2^{256} \approx 10^{77}$$

Given the computer system as above, the time to decode the message is:

$$\frac{10^{77}}{(2*10^{18})} = 5.10^{63} \text{ seconds} \approx 1.6*10^{56} \text{ years.}$$

Obviously, the security is greatly strengthened in comparison with the Advance Elgamal algorithm.

The author has also made a comparison of the encoding and decoding speed of the proposed algorithm (1) and the RSA one. The experimental condition is a computer with the configuration: CPU – Intel (R) Core (TM) i7-9750H, 2.60 GHz, RAM 8.00 GB, Windows 10 operating system. Two algorithms are implemented on the Java language with data to be encrypted is 16 bytes in size. The results are presented in Table 2.

Table 2 Comparing the encoding and decoding time of RSA algorithm and proposed algorithm

Encoding time (milliseconds)		*Decoding time (milliseconds)*	
RSA	*Proposed algorithm*	*RSA*	*Proposed algorithm*
759	121	746	116

As can be clearly seen in Table 2, the encoding and decoding time of the proposed algorithm are much better than RSA.

As for the evaluation of security and the speed of encryption and decryption, it can be claimed that the proposed algorithm (1) is appropriate for online information transmission. The implementation of algorithm (1) in a stratified information transmission system with intermediate information transmission layer has enhanced the security while still ensuring the real-time information transmission speed.

4 Conclusions

In this research, the authors have improved the asymmetric encryption algorithm based on GA, which increases the quality and efficiency in information security. In parallel with that is the implementation of a stratified information transmission architecture that enhances the confidentiality and authenticity of information, ensures the safety of information in the transmission process, eliminates the lost or falsification of information caused by cyber-attacks. Based on the proposed algorithms, the authors have developed and put into experimental implementation at the VNMAC, meeting the current operations in real-time transmission as well as the accuracy of the information.

Through empirical evaluation, the proposed algorithm has successfully demonstrated its security and processing speed. The research results are probably applicable in various different key encryption systems and continue to be upgraded to better the execution speed. The results can be applied across multiple security systems, in commercial transactions as well as the creation and authentication of digital signatures.

References

Alhussain, A.H. (2015) 'Key exchange based on genetic algorithm', *International Journal of Scientific Research in Science, Engineering and Technology*, Vol. 1, No. 1, pp.57–61.

Ali, S.T. and Murray, J. (2016) *An Overview of End-to-End Verifiable Voting Systems*, ArXiv, abs/1605.08554 [online] https://arxiv.org/abs/1605.08554 (accessed 15 February 2020).

Baykara, M., Daş, R.s. and Tuna, G. (2017) 'A novel symmetric encryption algorithm and its implementation', *Turkish Journal of Science & Technology*, Vol. 12, No. 1, pp.5–9.

Das, S.B., Mishra, S.K. and Sahu, A.K. (2020) 'A new modified version of standard RSA cryptography algorithm', in Elçi, A., Sa, P.K., Modi, C.N., Olague, G., Sahoo, M.N. and Bakshi, S. (Eds.): *Smart Computing Paradigms: New Progresses and Challenges*, pp.281–287, https://doi.org/10.1007/978-981-13-9680-9_24.

Dossogne, J. and Lafitte, F. (2015) 'Blinded additively homomorphic encryption schemes for self-tallying voting', *Journal of Information Security and Applications*, Vol. 22, pp.40–53, https://doi.org/10.1016/j.jisa.2014.07.002.

Fang, Y., Cong, L. and Deng, J. (2019) 'Research and design of an improved ElGamal digital signature algorithm', *IOP Conference Series: Materials Science and Engineering*, Vol. 569, No. 5, p.052041, https://doi.org/10.1088/1757-899X/569/5/052041.

Karabey, I. and Akman, G. (2016) 'A cryptographic approach for secure client – server chat application using public key infrastructure (PKI)', *11th International Conference for Internet Technology and Secured Transactions (ICITST-2016)*, pp.442–446, https://doi.org/10.1109/ICITST.2016.7856750.

Kucharczyk, M. (2010) 'Blind signatures in electronic voting systems', in Kwiecień, A., Gaj, P. and Stera, P. (Eds.): *Computer Networks*, pp.349–358, https://doi.org/10.1007/978-3-642-13861-4_37.

Kumar, R., Tayal, A. and Kapil, S. (Eds.) (2018) 'Analyzing the role of risk mitigation and monitoring in software development', https://doi.org/10.4018/978-1-5225-6029-6.

Marz, N. and Warren, J. (2015) *Big Data: Principles and Best Practices of Scalable Real-time Data Systems*, Manning, Shelter Island, NY.

Naik, P. and Naik, G. (2014) 'Symmetric key encryption using genetic algorithm', *National Conference on Innovations in IT and Management*, 1 February, pp.1–5, Sinhgad, Pune, India [online] https://www.researchgate.net/publication/269757592_Symmetric_Key_Encryption_using_Genetic_Algorithm (accessed 20 February 2020).

Nekoei, M., Mohammadhosseini, M. and Pourbasheer, E. (2015) 'QSAR study of VEGFR-2 inhibitors by using genetic algorithm-multiple linear regressions (GA-MLR) and genetic algorithm-support vector machine (GA-SVM): a comparative approach', *Medicinal Chemistry Research*, Vol. 24, No. 7, pp.3037–3046, https://doi.org/10.1007/s00044-015-1354-4.

Okeyinka, A.E. (2015) 'Computational speeds analysis of RSA and ElGamal algorithms on text data', *Proceedings of the World Congress on Engineering and Computer Science 2015*, San Francisco, USA, 21–23 October, Vol. 1, pp.115–118.

Okeyinka, A.E. (2017) 'Computational complexity study of RSA and ElGamal algorithms', in Ao, S-I., Kim, H.K. and Amouzegar, M.A. (Eds.): *Transactions on Engineering Technologies*, pp.233–243, https://doi.org/10.1007/978-981-10-2717-8_17.

Oluwatosin, H.S. (2014) 'Client-server model', *IOSR Journal of Computer Engineering*, Vol. 16, No. 1, pp.57–71, https://doi.org/10.9790/0661-16195771.

Pham, H.L. and Nguyen, T.S. (2006) 'Advanced Elgamal asymmetric key model in the future', *Journal of Science and Technology*, Vol. 44, No. 2, pp.1–5.

Rao, R.V. and Selvamani, K. (2015) 'Data security challenges and its solutions in cloud computing', *Procedia Computer Science*, Vol. 48, pp.204–209, https://doi.org/10.1016/j.procs.2015.04.171.

Shaktawat, R., Shaktawat, R.S., Lakshmi, N., Panwar, A. and Vaishnav, A. (2020) 'A hybrid technique of combining AES algorithm with block permutation for image encryption', *Reliability: Theory & Applications*, Vol. 15, No. 1, pp.51–56.

Sivakumar, T.K., Sheela, T., Kumar, R. and Ganesan, K. (2017) 'Enhanced secure data encryption standard (ES-DES) algorithm using extended substitution box (S-Box)', *International Journal of Applied Engineering Research*, Vol. 12, No. 21, pp.11365–11373.

Sokolov, A.V. and Shangin, V.F. (2002) *Zashchita informatsii v raspredelennykh korporativnykh setyakh i sistemakh [Information Security in Distributed Corporate Networks and Systems]*, in Russian, DMK Press, Moscow.

Verma, A., Guha, P. and Mishra, S. (2016) 'Comparative study of different cryptographic algorithms', *International Journal of Application or Innovation in Engineering & Management*, Vol. 5, No. 2, pp.58–63 [online] https://www.ijettcs.org/pabstract.php?vol=Volume5Issue2&pid=IJETTCS-2016-03-24-29 (accessed 16 March 2020).