2019 IEEE 19th International Conference on Communication Technology

Implementation of Encryption and Decryption Algorithms for Security of Mobile Devices

B V Varun
ECE Department
PES UNIVERSITY
Bangalore, India
e-mail: varunbv95@gmail.com

Abhishek M V
ECE Department
PES UNIVERSITY
Bangalore, India
e-mail: abhishekmv11@gmail.com

Akshay Chanabasappa Gangadhar
ECE Department
PES UNIVERSITY
Bangalore, India
e-mail: akshay220197@gmail.com

Purushotham U
ECE Department
PES UNIVERSITY
Bangalore, India
e-mail: purushothamu@pes.edu

***Abstract*—Progress of mobile communication and VLSI technology has aided in development of smart devices. These devices process the information of various formats and sizes in a limited amount of time. This information will be either stored in the devices or in cloud, hence there is a need for some kind of methodology to process and secure the data. Implementation of new algorithms to secure the information is always of immense interest. These algorithms will improve the performance of smart devices and helps for better human-machine interaction. Generally, symmetric and asymmetric approaches are used to secure the data from unauthorized users or attacks. Considering the amount of delay and complexity involved in processing the data, various forms of algorithms are used. In this paper, we propose a novel algorithm to secure the data from vulnerable attacks. These algorithms can be implemented on various platforms. The experimental results demonstrate an improvement of 10% for contacts and 15% for the encryption of images as compared to other conventional approaches.**

Keywords-VLSI; smart devices; cloud; secure; algorithms; symmetric and asymmetric approaches; delay; complexity

I. INTRODUCTION

In the cutting edge situation, a lot of information is created by different devices; this information goes straightforward through smart instruments, for example, smart watches personal devices, mechanized vehicles, and so forth. This enormous amount of information should be stored for any future usage. It is necessitated that this information is continually transmitted between different approved clients. This information may contain certain private data which might be identified with an individual or an association.

In the present world, this information is under consistent risk of being misused by any malevolent clients [5]. This necessitates any secret information is verified amid capacity and transmission between different clients [10]. Different encryption algorithms are utilized to improve the security for these information [1]. These algorithms guarantee that any classified information isn't utilized by unapproved clients.

Conventional symmetric and asymmetric algorithms such as AES [12] and RSA [1] provide security to some extent. Some of the attacks such as man in the middle attack, traffic analysis, masquerading makes these algorithms more susceptible to threats [3, 4]. The information or data processed in cloud [15, 16] is more vulnerable for attacks. Therefore, it is very much necessary to use appropriate algorithms for different forms of information (documents, images).

Subsequently, we are presently proposing algorithms to encode and decode different kinds of information. Three different algorithms are proposed to provide security for contact, documents and images. These algorithms provide security while storing and during transmission over the channel.

II. DATA SECURITY

Network security is a study of various policies and methodologies to be adopted for preventing any unauthorized access or modification, denial of service or misuse of a computer Network. Network security consists of a network Administrator whose primary objective is to provide authorization of access to data in the network. Every user is assigned some authenticating information such as a login ID or a username and a password [13, 14] which gives access to the information and other authorized resources within a network.

Data security refers to various privacy measures that are undertaken by an individual or an organization. These measures or policies prevent any computer or database or any websites from being accessed by the unauthorized users. It mainly deals with providing security to data against malicious usage. Data security plays an important role in every information technology related fields of any organization. Any data related to an organization has to be

978-1-7281-0535-2/19/$31.00 ©2019 IEEE

secured in order to prevent any loss or damage to the organization.

III. METHODOLOGY

Three algorithms are developed to provide security for information stored in the mobile phone. The case A, B discusses about encryption and decryption algorithms for the information stored as contact list. The case C, D discusses about encryption and decryption algorithms for the information stored as documents in any forms. The case E, F discusses about encryption and decryption algorithms for images.

A. Encryption Algorithm for Contact

Input Data : 8861298558
Step 1: Divide the ten digit contact into blocks of 2 digits each. Swap the 1st block with the 5th and 2nd block with the 4th. Swap the digits within the middle block.
5885926188
Step 2: Subtract every digit within the contact from the number 9.
4114073811
Step 3: Divide the contact into two halves comprising of 5 digits and swap the halves.
7381141140
Step 4: Consider each half. Swap the first two digits with the last two digits and retain the middle digit in the same position. Perform similar operation on the other half as well.
1187340141
Step 5: Convert into bits and perform various bit operations as follows. **0001 0001 1000 0111 0011 0100 0000 0001 0100 0001**
Take the compliment of every bit obtained. **1110 1110 0111 1000 1100 1011 1111 1110 1011 1110**
Rearrange the bits in such a way that 1st bit of all the blocks are grouped together, then the 2nd bits are grouped together and so on. **1101 1111 1111 1010 1101 1110 0111 1100 1001 1010**
Consider block of 4 digits each. Within every block, retain 1st and 3rd bits as it is, XOR 2nd and 4th bits with 1st and 3rd respectively **1001 1010 1010 1111 1001 1011 0110 1000 1101 1111**
Step 6: Generate a random 40 bit key. XOR this key with the 40 bit encrypted contact obtained above. The above obtained result is the final encrypted contact which can be transmitted over an open channel.
The above output is XOR'ed with the 40 bit randomly generated key.

B. Decryption Algorithm for Contact

Decryption is the reverse process of encryption. The encrypted copy of the contact is received and decrypted using the following steps.

Step 1: The obtained contact is XOR'ed with the randomly generated 40 bit key which is transmitted along with the encrypted contact.

Step 2: Perform various bit operations as explained below.

- Divide the obtained contact into blocks of 4 bits each. Keeping 1st and 3rd bit as it is, XOR 2nd and 4th bits 1st and 3rd bits respectively.
- Rearrange the bits in such a way that the first 10 bits are placed in the first position of all the ten blocks, the next 10 bits and so on.
- Take the compliment of all the bits obtained.

Step 3: convert the bits into decimal form. Perform swap operation on each half of the contact. Swapping is done is such a way that first two bits are swapped with last two bits but retain the middle bit. Perform similar operation on the other half as well.

Step 4: Swap the two halves thus obtained.

Step 5: Subtract every digit from the number 9.

Step 6: Divide the number into two digits each. Swap 1st block with 5th block, 2nd block with 4th block. Swap the bits within the middle block.

The contact thus obtained in the above step is the decrypted contact which can be used for further communication purposes.

Any document consists of large amount of data. In this algorithm, this data is divided into blocks of ten characters each. Every block is given as an input to the encryption algorithm and all the blocks are encrypted simultaneously. Every character in the block forms a sub block within the block.

C. Encryption Algorithm for Documents

Any document consists of large amount of data. In this algorithm, this data is divided into blocks of ten characters each. Every block is given as an input to the encryption algorithm and all the blocks are encrypted simultaneously. Every character in the block forms a sub block within the block.

E n c r y p t i o n
Step 1: All the characters in the block are converted into their respective ASCII values.
69 110 99 114 121 112 116 105 111 110
Step 2: These values in the block are then rearranged based on a substitution table which is loaded onto the algorithm.
121 116 112 99 69 114 110 110 105 111
Step 3: A random value X in the range 0<X<126 is added to every value in the block. This value of X acts as the primary key for the encryption algorithm. All the values in the block are checked for values greater than 127. If any value is greater than 127, then cyclic rotation method is applied to those values.
136 131 127 114 84 129 125 125 120 126 ↓ **9 4 0 114 84 2 125 125 120 126**
Step 4: Converting into bits and perform various bit operations as follows. **0000 0000 0000 0111 0101 0000 0111 0111 0111 0111** **1001 0100 0000 0010 0100 0010 1101 1101 1000 1110**
Take compliment of every bit in the block. **1111 1111 1111 1000 1010 1111 1000 1000 1000 1000** **0110 1011 1111 1101 1011 1101 0010 0010 0111 0001**
Rearrange the above obtained bits in such a way that 1st bit of every sub block is grouped together and then 2nd bits and so on. **1111 1111 0000 1100 1001 0111 0010 0010 1011 1111**

1111 1001 1110 0011 0000 1100 1101 1110 1001 0011
Retain the 1st, 3rd, 5th and 7th bits of every sub block, XOR them with 2nd, 4th, 6th and 8th bits respectively.
1010 1010 0000 1000 1101 0110 0011 0011 1110 1010
1010 1101 1011 0010 0000 1000 1001 1011 1101 0010

The above obtained result of every block is the encrypted copy of that block. Similarly every block of data is encrypted in a sequential manner and transmitted. The primary key used in the encryption mechanism is also transmitted in the encrypted form to the other user.

D. Decryption Algorithm for Documents

Each block of data is received and given as an input to the decryption algorithm in a sequential manner. The received data is divided into 10 sub blocks of eight bits each.

Step 1: Perform various bit operations as explained below.

- Retain the 1st, 3rd, 5th and 7th bits of every sub block, XOR them with 2nd, 4th, 6th and 8th bits respectively.

Rearrange the bits in such a way that first 10 bits of the bit stream are placed in the first position of every sub block and next 10 bits are placed in the second position and so on.

- Take compliment of every bit in every block.

Step 2: All the values in the sub blocks are converted into decimal values. The values X is subtracted from all the values in the sub blocks. The value 127 is added to only those values which are found to be negative.

Step 3: The above obtained values are rearranged based on the reverse substitution table loaded onto the decryption algorithm.

Step 4: Rearranged values are converted back to their respective characters to form the document.

All the blocks are obtained and decrypted similarly in a sequential manner and rearranged to form the whole document at the receiver.

E. Encryption Algorithm for Images

Every image is represented in terms of pixels. Any image is stored in the form a matrix with every element corresponding to a pixel.

Step 1: Convert any given image into standard 512x512 image. This image is stored in the form a 512x512 matrix. Add a random value X in the range 0<X<254 to only those columns in the matrix with odd number indexing. Perform cyclic rotation only those values exceeding the value 255.

Step 2: Now add the value X+1 to only those rows with even number indexing in the matrix. Perform cyclic rotation on these values as well.

Step 3: Perform swapping of rows in such a way that first row is swapped with the last row, second row with the row previous to last row and so on. Repeat the similar operation columns as well.

Step 4: Convert every element in the matrix into bits. Retain the columns with odd number indexing, XOR them with columns immediately next to them and store them in these columns. Ex: Retain C1, XOR C1 with C2 and store the result in C2.

Step 5: Now, retain rows with even number indexing, XOR them with rows immediately above them and store the result in those rows. EX: Retain R2, XOR R2 with R1 and store the result in R1.

Step 6: Take the compliment and transpose of the above obtained matrix.

The matrix obtained after the encryption algorithm is divided into a number of packets based on the maximum packet size and transmitted over the channel. These packets are transmitted along with the header which indicates the order in which these packets have to be rearranged at the receiver.

F. Decryption Algorithm for Images

All the packets are received and rearranged based on the headers. These packets are given together as an input to the decryption algorithm.

Step 1: The above obtained matrix is in the binary form. Take the compliment and transpose of the above matrix.

Step 2: Retain the rows with even number indexing, XOR them with rows immediately above them and store the result in those rows. EX: Retain row R2, XOR R2 with R1 and store the result in R1.

Step 3: Now, retain columns with odd number indexing, XOR them with columns immediately next to them and store the result in those columns. Ex: Retain columns C1, XOR them C2 and store the result in C2.

Step 4: Convert the above bits into decimal form. Perform swap operation between columns in such a way that first column is swapped with last column, second column is swapped with the column previous to the last column and so on. Perform similar operation on rows as well.

Step 5: Consider only the rows with even number indexing. Subtract the value X+1 from all the values in these rows. Add the value 255 to the values which are found to be negative.

Step 6: Now, subtract the value X only from those values in the columns with odd number indexing. Add the value 255 to those values which are found to be negative.

The above obtained matrix is the decrypted copy of the image. This represents the image which was encrypted and sent over the channel from user 1.

IV. IMPLEMENTATION

The data to be transmitted is encrypted and sent over the channel from user 1 to the user 2 via the following steps.

Step 1: A TCP connection is established between the two users via a well defined port between the two users.

Step 2: The data to be transmitted is encrypted on the user 1's device by using the corresponding encryption algorithm.

Step 3: A well known key generation algorithm such as RSA algorithm is run on both the user's devices to generate their respective public and private key pairs.

Step 4: The public keys of both the user's are exchanged with each other via the TCP connection.

Step 5: The primary key used during the encryption of data is transmitted from user 1 to user 2 by encrypting it using the public key of user 2. This key can be decrypted at user 2 using his private key only.

Step 6: A second TCP connection is established between the two users via another port.

Step 7: The encrypted data to be transmitted is sent over this second TCP connection.

Step 8: The primary key is first decrypted using the private key of user 2 and is given as an input to the decryption algorithm.

Step 9: The encrypted data is decrypted using the corresponding algorithm and the key and stored for further usage on the user 2's device.

V. RESULTS

The algorithms are developed using python programming language and are executed on spyder software. 5.1 shows the input, output and randomly generated key for contacts for the proposed algorithms. 5.2 shows the input, output and random key for documents for the proposed algorithms. 5.3 shows the input, output and randomly generated key for images for the proposed algorithms.

5.1: CONTACTS:

Input: 8861298558

Random Generated Key: 629872373165

Output: d808cac572

5.2: DOCUMENT:

Input: Encryption

Random Key: 15

Output: ª,Ðh9;ią

5.3: IMAGES:

Input:

Random key: 121

Output:

Figure 1. Input for the image Encrytpion algorithm

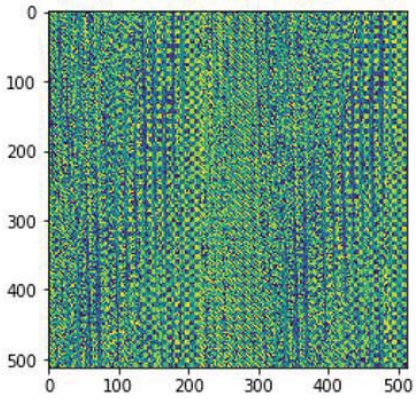

Figure 2. Encrypted image of the input

Comparing the results of the above two algorithms, it was observed that the three proposed algorithms in this paper were found to have an improvement in complexity of approximately 10% -15% .

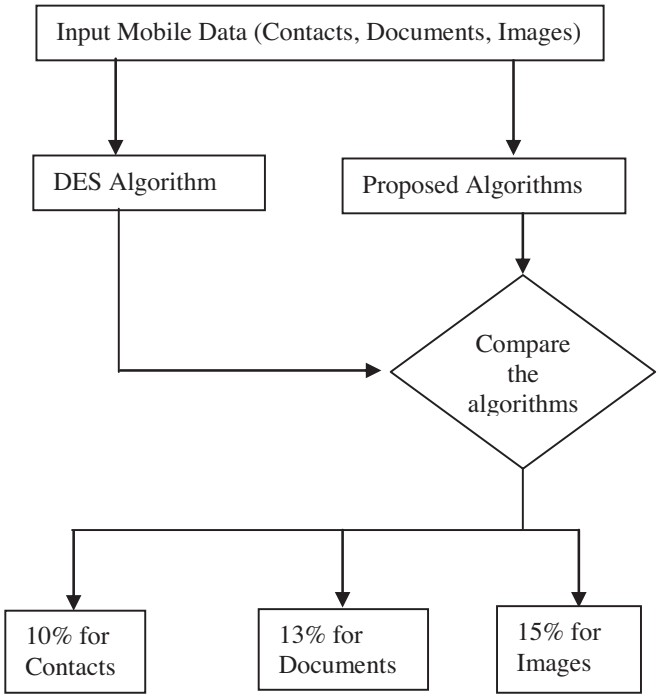

Figure 3. Comparing the developed algorithms with DES

-The complexity of the algorithm for contacts lies in the range 45% - 65%.

-The complexity of the algorithm for documents lies in the range 45% - 60%.

-The complexity of the algorithm for images lies in the range 50% - 65%.

VI. CONCLUSION

A large amount of information is generated on a daily basis from most of the devices. This data is required to be

stored as well as transmitted between various users. It can also contain some amount of confidential data which needs to be secured. The conventional approaches such as AES and RSA have been proposed to provide security for such data. A number of attacks have been discovered which make these algorithms vulnerable for threats. The above proposed algorithms are observed to provide considerable amount of security to different kinds of data such as contacts, documents and images. The complexity of the above algorithms is checked and is found to be better than the conventional methods.

REFERENCES

[1] R.L. Rivest, A. Shamir, and L. Adleman “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”.

[2] James F Kurose and Keith W Ross “Computer Networking: A Top Down Approach”Sixth edition pearson edition 2013.

[3] William Stalling “Cryptography and network security”Pearson education edition 2013

[4] Behrouz A Forouzan and Debdeep Mukhopadhyay “Cryptography and Network Security” TMH publications 2nd edition.

[5] X. Jing, Z. Yan and W. Pedrycz, "Security Data Collection and Data Analytics in the Internet: A Survey," in IEEE Communications Surveys & Tutorials, vol. 21, no. 1,pp.586-618,Firstquarter2019.doi: 10.1109/COMST.2018.2863942

[6] Z. A. Balouch, M. I. Aslam and I. Ahmed, "Energy efficient image encryption algorithm," 2017 International Conference on Innovations in Electrical Engineering and Computational Technologies (ICIEECT), Karachi, 2017, pp. 1-6. doi: 10.1109/ICIEECT.2017.7916541

[7] R. K. Singh, T. Begum, L. Borah and D. Samanta, "Text encryption: Character jumbling,"2017 International Conference on Inventive Systems and Control (ICISC), Coimbatore, 2017, pp. 1-3. doi: 10.1109/ICISC.2017.8068691

[8] G. L. Prakash, M. Prateek and I. Singh, "Data encryption and decryption algorithms using key rotations for data security in cloud system," 2014 International Conference on Signal Propagation and Computer Technology (ICSPCT 2014), Ajmer, 2014, pp. 624-629.

[9] R. Yu et al., "Authentication with Block-Chain Algorithm and Text Encryption Protocol in Calculation of Social Network," in IEEE Access, vol. 5, pp. 24944-24951, 2017.doi: 10.1109/ACCESS.2017.2767285

[10] N. Veeraragavan, L. Arockiam and S. S. Manikandasaran, "Enhanced encryption algorithm (EEA) for protecting users' credentials in public cloud," 2017 International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET), Chennai, 2017, pp. 1-6. doi: 10.1109/ICAMMAET.2017.8186644

[11] Q. Chen, Q. Chen, M. Yao and J. Mo, "Design of encryption algorithm of data security for Wireless Sensor Network," 2011 International Conference on Electrical and Control Engineering, Yichang, 2011, pp. 2983-2986. doi: 10.1109/ICECENG.2011.6057724

[12] Z. Yang, A. Li, L. Yu, S. Kang, M. Han and Q. Ding, "An Improved AES Encryption Algorithm Based on Chaos Theory in Wireless Communication Networks," 2015 Third International Conference on Robot, Vision and Signal Processing (RVSP), Kaohsiung, 2015, pp. 159-162. doi: 10.1109/RVSP.2015.45

[13] M. H. Ahmadzadegan, A. Khorshidvand and M. Pezeshki, "A method for securing username and password against the Keylogger software using the logistic map chaos function," 2015 2nd International Conference on Knowledge-Based Engineering and Innovation (KBEI), Tehran, 2015, pp. 1071-1073. doi: 10.1109/KBEI.2015.7436194

[14] M. Shahid and M. A. Qadeer, "Novel scheme for securing passwords," 2009 3rd IEEE International Conference on Digital Ecosystems and Technologies, Istanbul, 2009, pp. 223-227. doi: 10.1109/DEST.2009.5276738

[15] S. Ramgovind, M. M. Eloff and E. Smith, "The management of security in Cloud computing," 2010 Information Security for South Africa, Sandton, Johannesburg, 2010, pp. 1-7. doi: 10.1109/ISSA.2010.5588290

[16] B. R. Kandukuri, R. P. V. and A. Rakshit, "Cloud Security Issues," 2009 IEEE International Conference on Services Computing, Bangalore, 2009, pp. 517-520. doi: 10.1109/SCC.2009.