Implementation of Encryption and Decryption Algorithms for Security of Mobile Devices

TEAM MEMBERS:
AASTHA GADHVI:60003210132
VIRTI SHAH:60003210119

***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 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 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.

B. 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: IMAGES:

Input:

+ Code + Text

```
import numpy as np
from google.colab.patches import cv2_imshow
import cv2

def resize_image(image, width, height):
    return cv2.resize(image, dsize=(width, height), interpolation=cv2.INTER_AREA)

def encrypt_image(image, key):
    image = resize_image(image, 512, 512).astype(np.uint8)

    image[:, 1::2] = (image[:, 1::2] + key) % 256
    image[1::2, :] = (image[1::2, :] + (key + 1)) % 256
    image = image[::-1, ::-1]

    for col in range(0, image.shape[1] - 1, 2):
        image[:, col] ^= image[:, col + 1]

    for row in range(0, image.shape[0] - 1, 2):
        image[row, :] ^= image[row + 1, :]

    return image

def decrypt_image(image, key):
    for row in range(0, image.shape[0] - 1, 2):
        image[row, :] ^= image[row + 1, :]

    for col in range(0, image.shape[1] - 1, 2):
        image[:, col] ^= image[:, col + 1]

    image = image[::-1, ::-1]
```

+ Code + Text

```
    image[:, 1::2] = (image[:, 1::2] + key) % 256
    image[1::2, :] = (image[1::2, :] + (key + 1)) % 256

    for col in range(0, image.shape[1] - 1, 2):
        image[:, col] ^= image[:, col + 1]

    for row in range(0, image.shape[0] - 1, 2):
        image[row, :] ^= image[row + 1, :]

    return image

def decrypt_image(image, key):
    for row in range(0, image.shape[0] - 1, 2):
        image[row, :] ^= image[row + 1, :]

    for col in range(0, image.shape[1] - 1, 2):
        image[:, col] ^= image[:, col + 1]

    image[:, 1::2] = (image[:, 1::2] - key) % 256
    image[1::2, :] = (image[1::2, :] - (key + 1)) % 256

    return image

def stackImages(scale, imgArray):
    rows = len(imgArray)
    cols = len(imgArray[0])
    rowsAvailable = isinstance(imgArray[0], list)
    width = imgArray[0][0].shape[1]
    height = imgArray[0][0].shape[0]
    if rowsAvailable:
        for x in range (0, rows):
```

Random key: 121

Output:

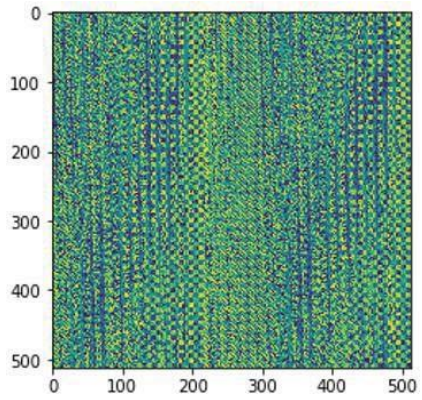

Figure 2. Encrypted image of the input

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%

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.

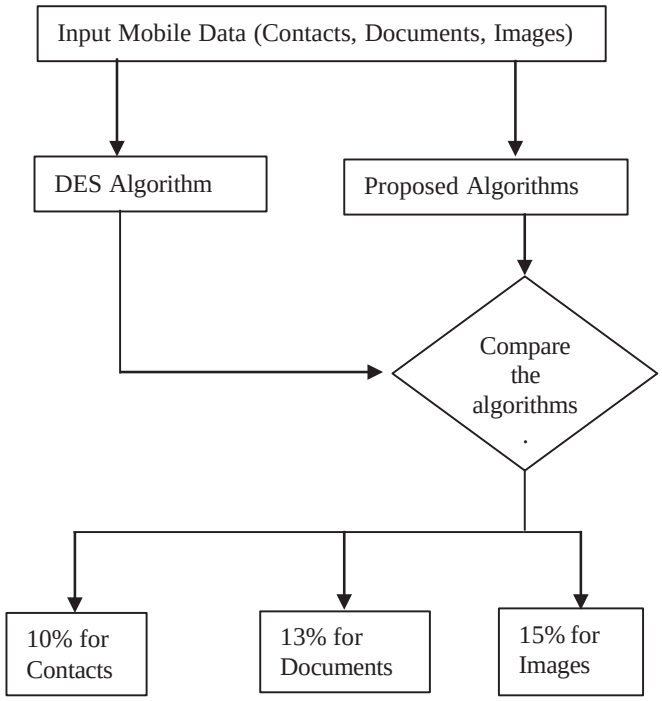

Figure 3. Comparing the developed algorithms with DES

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.