Journal of Theoretical and Applied Information Technology
30th September 2018. Vol.96. No 18

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

# TRIO-SECURITY MODEL FOR SECURING DATA OF FILE SHARING IN MOBILE ENVIRONMENT

**[1]SITI RAHAYU SELAMAT, [2]S.L. MUHAMMAD HAFIZUDDIN, [3]ZAKIAH AYOP, [4]ROBIAH YUSOF**

[1, 3, 4]Senior Lecturer and Researcher, Universiti Teknikal Malaysia Melaka, Malaysia

[2]Researcher, Universiti Teknikal Malaysia Melaka, Malaysia

E-mail: [1]sitirahayu@utem.edu.my, [2]daen210495@gmail.com, [3]zakiah@utem.edu.my, [4]robiah@utem.edu.my

**ABSTRACT**

Nowadays, mobile phone is a sharing medium to connect to one's own social network and larger society for its convenience and high-speed. Consequently, data breach incidents by unauthorized parties might often occur. Therefore, the aim of this study was to secure sensitive information in the CIA (Confidentiality, Integrity, Availability) triad. Data Security model, Trio-Security, was integrated within file sharing application as a solution to the problem as this model used three different integrated technologies namely Message Digest, Cryptography and Steganography to provide security for the CIA data. The most suitable and compatible algorithm for each technology was used for mobile environment specifications. The results of this project determined the quality of the output for each algorithm. Based on the results, the integration of Trio-Security model with the file sharing application was able to increase the security level when transferring sensitive data.

**Keywords:** *Trio-Security, File Sharing, Mobile, Cryptography, Steganography, Message Digest*

## 1. INTRODUCTION

Currently, data breach is very serious and undergoing a rapid growth over the years. Even though there are many sources of data breach incidents, only two sources have been continuously increasing over the past 6 years; Malicious Outsider and Hacktivists [2]. This scenario indicates Men-In-The-Middle, one of the cyber-attacks, is involved where the cybercrime process steals the data transmitted across the network.

Currently, literature in data security model for the current transferring method in a mobile environment lacks the standard security level of data protection. The model is only focusing on the confidentiality of the data through cryptography technology. Even though strong cryptography algorithm provides high level of data security, acknowledging the attacker that the data sent is encrypted information will increase their curiosity and entice them to decrypt the information to recover the original information.

Additionally, with the rapidly emerging of super computer over the years, the probability of the unauthorized decryption process to succeed is increasing. Hence, a model that can provide the CIA Triad (Confidentiality, Integrity and Availability) and conceal the presence of the data is needed. Therefore, this project proposed Data Security Model known as Trio-Security. This model would assist in securing the confidentiality of data, providing the integrity of the data and ensuring the real information would not be available to the unauthorized users by concealing the presence of the data within other non-sensitive information. Thus, data security is enhanced when transmitted across the network by integrating the Message Digest, Cryptography and Steganography technology.

Message Digest is a cryptographic hash function that represents building block. This technology provides the data integrity which offers certainty to the receiver that the received data is the same as the ones sent by the sender [3] and can represent the signature value of the sensitive data [4]. Cryptographic is developed to guarantee the security of information by securing and maintaining the information's confidentiality value [5]. This technology will perform many different transformations and substitutions on the data structure known as original data and transform into a scrambled message known as cipher data [6].

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Steganography is a technique that ensures the availability of the data which is only meant for the authorized receiver by hiding the data inside a carrier medium [7]. Multimedia items such as video, text, audio, images and other digitally representative code are the most suitable items to be used as carriers to hold the hidden information known as stego-carrier [8][9].

## 2. TRIO-SECURITY MODEL ARCHITECTURES

Trio-Security model provides description of data security process and its activity for integrating three security technologies namely message digest (*Md*), cryptography (*Cr*) and steganography (*St*) in a mobile environment to secure the data while being transmitted across the network. This model consists of objects, techniques and outputs.

### 2.1 Objects

The objects represent entities which are needed as parts of the data security process. These entities include information in any types of format which will be transferred across the network. Entities discovered in data security process have many different formats with different responsibilities. The responsibility varies depending on the sensitivity of the information. There are two types of objects; Sensitive and Non-sensitive objects. Sensitive objects represent information with various formats that carry sensitive information. These objects are the information chosen by users to be transmitted across the network to the receiver destination. Hence, this object will undergo the data security process to secure its value of confidentiality, integrity and availability. For example, sensitive object in this model can be in any format of electronic information such as multimedia items such as image, video and audio or any other software processing information such as application, Words and PDF. Non-sensitive objects represent information which carries no sensitivity value. Hence, this information has been allowed to be disclosed by an unauthorized user. A crucial element of this object is that it does not make unauthorized users to be suspicious of the presence of other information within this information. The non-sensitive information can be in the form of any multimedia items such as image, audio and text. For example, non-sensitive object in this model will be in image format where valuable information is to be transmitted across the network but not necessary to be sensitive information.

Based on the categories above, both objects are chosen by a user before the transmitting process occurs. The significant difference between the objects is that only sensitive object will undergo data security process while the non-sensitive data will act as disguise carriers that hide sensitive data which are then, transmitted across open network.

### 2.2 Techniques

Techniques can be defined as the hierarchy of different security process that will be carried out by the objects in order to secure information. The level of security of the information relies on the choice of these techniques that will be integrated together and the capability of the techniques to be executed successfully within the mobile environment with certain specifications and limitations. There are three types of techniques namely Message Digest, Cryptography and Steganography. Message Digest technique represents a process to provide integrity value to the information by generating a hash value which indicates the signature of the sensitive information. Additionally, different information will have different hash values. Therefore, the integrity value of information is preserved by utilizing this technique. However, a few elements of this technique need to be verified in order to ensure the hash value generated is suitable in size and reliable where different information will never generate the same hash value when undergoing this technique.

Cryptography technique represents a process to secure confidentiality value to the information by scrambling the information structure into a different structure which carries unintelligence information [10]. There are many available algorithms related to this technique that have different specifications and architectures. Therefore, the algorithm is to be evaluated to be compatible with mobile environment and provide the best data security level to sensitive information. The criteria should be evaluated in terms of process round, processing speed, successful attacks and security level.

Steganography technique represents a process to secure the availability value to the information by concealing the presence of the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

information inside other non-sensitive information [7]. Although this model is specific only to image as information carrier, there are many available algorithms related to this technique that have different specifications and architectures [11]. Therefore, the algorithms should be evaluated to be compatible with mobile environment and conceal the presence of successful information so that it remains unknown from attackers [12]. The criteria namely transparency, robustness, payload capacity, processing performance and network performance should be evaluated.

The criteria are to be evaluated in order to choose the best and most suitable algorithm within each technique regarding the mobile environment and the quality and level of data security provided by the algorithm. By studying the stated criteria for each technique, a developer will identify the best algorithm available to be implemented in data security architectures. Basically, the relationship among the three techniques in Trio-Security model can be defined as in Equation 1

$$\text{TrioSecurity} = \{\text{Tech}\} = \{Md + Cr + St\} \quad (1)$$

### 2.3 Outputs

Outputs are the final result of the occurrence of an object and technique integrated together in certain rules and procedures. Each technique will manipulate the object and produce output whereas each technique will deliver different outputs. The outputs, then, are reconstructed to produce a single output which is transmitted across the network. For each output, specific attributes may be used to act as a unique value to trigger the action of the technique process. Additionally, a specific input is required for the process to be carried out successfully. Three outputs are produced from the technique process in this study; hash value, unintelligence information and stego-image. Hash value output, *Hv*, is an output produced by the Message Digest technique. This technique requires an input of a sensitive object to produce an output. Hash value is a unique signature value representing a sensitive object in which to ensure the integrity of the object that is sent across the open network. The hash value generated in this model will be used for comparison with the hash value of the received object at the recipient side. If the hash value remains the same, the integrity of the object is secured.

Unintelligence information output, *Ui*, is an output produced by the cryptography technique. This technique requires an input of a sensitive object and a secret key to produce an output. The unintelligence information is a reassembled structure of a sensitive object which carries no meaningful information. Hence, this information is not readable by human or appropriate reader software application which secures the confidentiality of a sensitive object. Stego-image output, Si, is an output produced by Steganography technique. This technique requires an input of the hash value, unintelligence information and non-sensitive object to produce an output [8]. The hash value and unintelligence information must be the output of the same sensitive object while the non-sensitive object is randomly or specifically chosen to hide the combination of the hash value and unintelligence information. Therefore, only a non-sensitive object will be sent across the open network, which secure the availability of the sensitive object as the object presence is concealed from unauthorized users. The outputs of *OBJ*, a sensitive object and *SOBJ* a non-sensitive object can be represented as in Equation 2.

$$Si(SOBJ)=\{SOBJ+(Ui(Hv(OBJ)+Ui(OBJ)))\} \quad (2)$$

## 3. EVALUATION OF TECHNIQUES ALGORITHM

Each of the technique in Trio-Security model has various different algorithms that offer different process requirements and architectures. Therefore, an analysis and evaluation on every available algorithm for each of the techniques is compulsory to determine the most suitable and compatible algorithm to be implemented within the system process in mobile environments.

### 3.1 Message Digest Algorithm

Four types of algorithms were available in Message Digest technology; Secure Hash Algorithm 0, Secure Hash Algorithm 1, Secure Hash Algorithm 2 and Secure Hash Algorithm 192. The criteria that should be evaluated were the size of the hash value and collision found. The chosen algorithm must have a reasonable size of hash value as the information was to be hidden inside the image carrier. Hence, unnecessary payload capacity for the image was to be avoided. Additionally, the algorithm must provide hash value without collision to ensure the produced hash value was reliable to secure the integrity of a sensitive object.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The comparison of the Secure Hash Algorithm is shown in Table 1.

*Table 1. Comparison Hash Algorithm [4]*

| Algorithm Name | Output Size | Rounds | Collision Found |
|---|---|---|---|
| SHA-0 | 160 | 80 | YES |
| SHA-1 | 160 | 80 | YES |
| SHA-2 | 256 / 384 / 512 | 64/80 | NO |
| SHA-192 | 192 | 64 | NO |

Based on Table 1, SHA-192 was the most suitable algorithm as it produced a hash value that was robust to collision and had the smallest size amongst the algorithms. However, literature showed that none of the hash algorithm is secured to ensure the integrity except SHA-2 [4]. Therefore, as the main purpose of this technique was to ensure the integrity of the data, SHA-2 with the output size of 256 was chosen to be the most suitable and compatible for the Trio-Security model.

### 3.2 Cryptography Algorithm

Various algorithms for cryptography technology that provide the same goal in which to secure the confidentiality of the data are available. However, after eliminating algorithms that had low level of security, the remaining algorithms for evaluation were DES, 3DES, AES, RSA, Blowfish and ECC. The criteria to be evaluated were processing round, processing speed, successful attacks and security level. The chosen algorithm must have less number of processing round, faster processing speed, high level of security, and if possible no successful attacks found to ensure this model used the most secure algorithm available with reasonable processing subjected to compatibility with mobile environment specification [10][13][14]. An analysis of the reviewed cryptographic algorithm is shown in Table 2.

Based on Table 2, Blowfish and ECC were the most suitable cryptography algorithms as both provided very high level of security, very fast processing speed, least number of rounds of process and no successful attack on Blowfish algorithm whereas only Doubling attack worked on ECC algorithm. The processing speed and number in round of process were necessary to make sure even with mobile specification, the cryptographic process could be executed faster without affecting the application performance).

*Table 2. Cryptography algorithm analysis*

| Algorithm Name | Round of Process | Processing Speed | Succeed Attacks Found | Level of Security |
|---|---|---|---|---|
| DES | 16 | Very Slow | Exclusive Key Search, Linear Cryptanalysis, Differential Analysis | Medium |
| 3DES | 48 | Very Slow | Related Key Attack | Medium |
| AES | 128, 192, 256 | Medium | Key Recovery Attack, Side Channel Attack | High |
| RSA | 1 | Medium | Brute Force Attack, Timing Attack | High |
| Blowfish | 16 | Very Fast | None | Very High |
| ECC | 1 | Very Fast | Doubling Attack | Very High |

### 3.3 Steganography Algorithm

As this model utilized Image Steganography technology, only algorithms with the respective specifications were taken into evaluation. The algorithms were LSB, Transform Domain (TD), Spread Spectrum (SS), Statistical Technique (ST), Distortion Technique (DT) and File & Pallet Embedding (FFE). The transparency, robustness and payload capacity are the criteria to determine whether the algorithm produces highest quality stego-image whereas processing speed is to determine the compatibility of the algorithm with the mobile specification and network performance effect to ensure the output of this algorithm does not affect the main purpose of the application in transferring data [9][12][15][16]. The analysis of the reviewed Image Steganography algorithm is shown in Table 3.

Based on Table 3, LSB algorithm was the most suitable as it only lacked in Robustness criteria where the stego-image was not robust

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

against any image manipulation processing such as compression and cropping.

*Table 3. Analysis of Image Steganography algorithm*

| Algo-rithm Name | Trans-parency | Robust-ness | Payload Capa-city | Proces-sing Speed | Affect Net. Perf. |
|---|---|---|---|---|---|
| LSB | ✓ | - | High | Fast | ✓ |
| TD | ✓ | ✓ | Low | Slow | - |
| SS | ✓ | - | Low | Slow | - |
| ST | ✓ | - | Low | Slow | - |
| DT | ✓ | - | Low | Slow | - |
| FFE | - | - | High | Slow | ✓ |

However, this algorithm produced an output that had high transparency value where the stego-image had no significant changes that could be detected by Human Visual System (HVS), high payload capacity, where an image carrier could hold a large amount of information with fast processing speed which was suitable for mobile specification and did not affect network performance as one image could hold large amount of information, so only a few images were to hold large amount of information. Therefore, bandwidth needed to be transmitted was not increased drastically.

## 4. DESIGN PROCESS OF TECHNIQUES ALGORITHM

### 4.1 Message Digest Process

The overall flow process of this technique is shown in Figure 1. Figure 1 shows the used object of this technique.

Based on Figure 1, the input required for this algorithm was a Sensitive object and the output was Hash value of the object. As the algorithm was using SHA-256, the hash value size was 256 bits. However, the output was already converted into Hexadecimal String to make it easy to be used in the upcoming process such as hiding and comparison process.

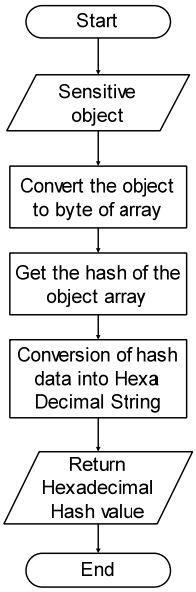


*Figure. 1. Flow Process of Message Digest Algorithm*

### 4.2 Cryptography Algorithm Process

This model used two types of algorithms which were Blowfish and ECC. The overall process of both algorithms are shown in Figure 2 and Figure 3 respectively. From the figures, the required inputs and the expected outputs of these algorithm process are presented.

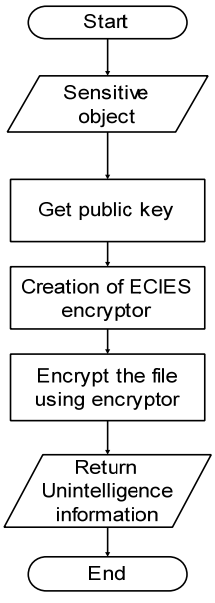


*Figure 2. Flow Process of ECC Algorithm*

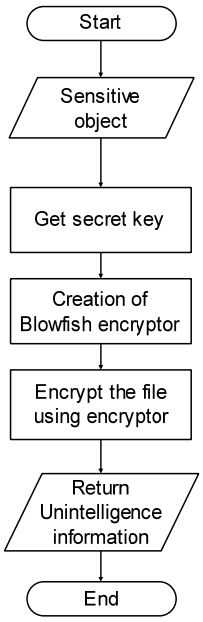


*Figure. 3. Flow Process of Blowfish Algorithm*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Based on Figure 2, inputs required for this algorithm were Sensitive object and ECC in Asymmetric Key encryption which also required Public Key of the receiver. An output of this algorithm was the Unintelligence information generated from scrambled data structure of the Sensitive object. In Figure 3, inputs required for this algorithm were Sensitive object and Blowfish in Symmetric Key encryption which also required Secret Key. The output of this algorithm was the Unintelligence information generated from scrambled data structure of the Sensitive object.

### 4.3 Steganography Algorithm Process

The steganography technique algorithm process was divided into two sub-processes; Initialize process and LSB Embedding process. The Initialize process was a process of reconstructing the information input and the non-sensitive object. Once the Initialize process was executed, the output of this process would be the input of the LSB Embedding process to produce the final output namely Stego-Image. The overall flow process of this technique is shown in Figure 4 and Figure 5. The figures showed the used object and the output of each process.

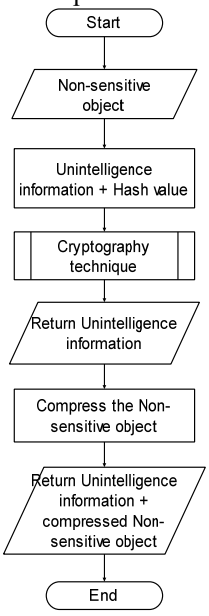


*Figure. 4. Flow Process of Initialize Process*

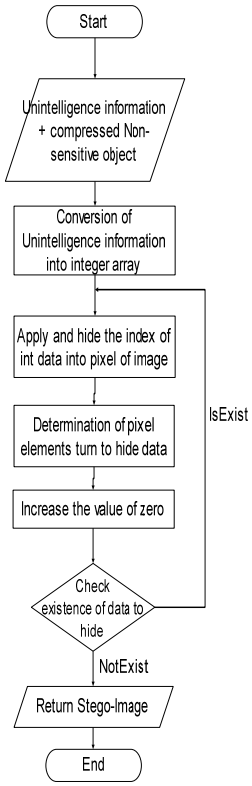


*Figure. 5. Flow Process of LSB Embedding Process*

Based on Figure 4, inputs required for this process were Non-sensitive object and the output of Message Digest technique and Cryptography technique. The outputs of these processes were the new Unintelligence information and compressed Non-sensitive object. Figure 5 showed the inputs required for this process resulting from the outputs of Initialize process; new unintelligence information and compressed non-sensitive object. The output of this process was Stego-Image.

## 5. OVERALL PROCESS OF TRIO-SECURITY MODEL

The overall Trio-Security process is shown in Figure 6. It shows the objects, techniques and outputs of the Data Security process of securing the information transmitted across the network in mobile environment. As illustrated in Figure 6, the process of securing data was initialized when both objects were selected. A sensitive object was selected by a user whereas non-sensitive object was randomly selected. Then, the process continued by undergoing three techniques in the hierarchy where the Message Digest was carried out first using the sensitive object then followed by cryptography using the same object. Second, the Steganography technique was executed using an output of the previous techniques and the non-sensitive object produced the final output, Stego-Image, by merging with the previous outputs. Finally, Cryptography technique was conducted and produced single unintelligence information which was embedded inside the non-sensitive information. In this study, as the implementation is developed using Native development method, hence the target is Android platform. As a result, it only accessible for user using smartphone with android operating system.

## 6. TESTING AND ANALYSIS ON FILE SHARING APPLICATION

The testing was a crucial process to ensure that the Data Security model (Trio-Security) provided all its functionality to the File Sharing application. The testing focused on an integration of the three techniques in Trio-Security model. Each technique was tested to ensure it was executed successfully on File Sharing application in a mobile environment.

### 6.1 Testing and Analysis on Message Digest Technique

The aim for this testing was to verify the capability of this technique to generate a hash value

Journal of Theoretical and Applied Information Technology
30th September 2018. Vol.96. No 18

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

from a file and the accuracy of the generated hash value to resist collision. The testing hash value procedure was derived from two main steps which verified the hash value generated and the accuracy of the hash value to avoid collision.

### 6.1.1 Verification of hash value generated

The verification of hash value generated was a process to verify whether algorithms used were able to generate a valid hash value representing sensible object. If an algorithm was designed correctly, it would generate the same hash value from the same sensible object. The comparison of hash value for one sensitive object that undergoes the same algorithm is shown in Figure 7.

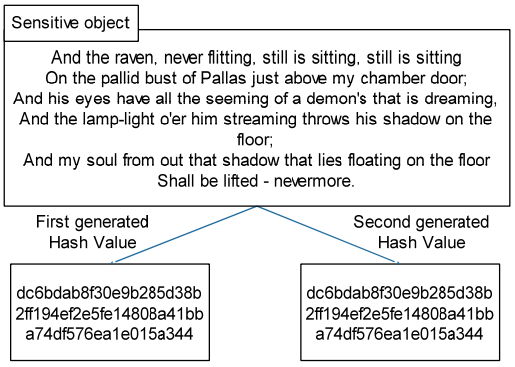


*Figure. 7. Comparison of two hash values for same file*

Based on Figure 7, the hash value generated for the file remained the same in both trials using the same algorithms. Therefore, the hash value generated by the algorithm was valid.

### 6.1.2 Verification of the accuracy of the hash value

The verification of hash value accuracy was a process to verify the reliability of the algorithm used to generate the hash value to avoid collision incident. Collision in hash value is an incident of two different files, generating the same hash value. This problem indicates that weak hash value is produced and easily deceived. This is because weak hash value is not reliable for different files may produce the same hash value and there is a probability that the integrity data is not secured as the application cannot identify which file has been modified as the hash value remains the same. The comparison of hash values between two different files with slight differences is shown in Figure 8. Based on Figure 8, the Message Digest algorithm was implemented in this project to generate a hash value that resisted to collision as the file had only slight differing of hash value generated. Hence, the hash value generated by the application was reliable to ensure the integrity of the file.

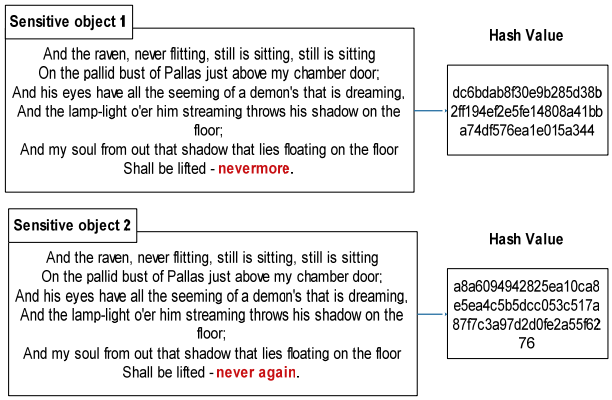


*Figure. 8. Comparison of two hash values for two slightly different file*

## 6.2 Testing and Analysis on Cryptographic Technique

The aim for this testing was to verify the reliability of this technique where the algorithms were successfully reassembled the sensitive object data structure into an unintelligence information data structure that was not readable by HVS and appropriate reader software applications. The testing cryptography procedure was derived from two main steps which verified the encryption status data and confidentiality status of encrypted data.

### 6.2.1 Verification of encryption status

This testing procedure was to verify the encryption process being carried out successfully using the generated public key. The verification process was executed by ensuring that the encryption process was successfully produced as an encrypted file. The encrypted file was created by the encryption process which is shown in Figure 9.


*Figure 9: The encrypted file create by encryption process*

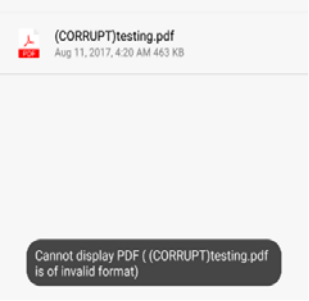


*Figure 10: The confidentiality status of encrypted file*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Based on Figure 9, the encryption process was successfully done as the encrypted file was successfully created. The encrypted file name would be added "(CORRUPT)" in the initial name to indicate the file could not be accessed as the file was corrupted. This might deceive the attacker to believe the file is not an encrypted file but a corrupted file which cannot be open.

### 6.2.2 Verification of the confidentiality status of encrypted data

This testing procedure was to verify whether the encrypted file was no longer an intelligent information which carried meaningful data which were similar prior to the cryptography process. This was a crucial testing procedure which indicated whether the encryption process was successfully performed its functionality or not. The status of encrypted file confidentiality is shown in Figure 10. In Figure 10, the encrypted file was no longer readable by HVS or an appropriate reader software application, indicating that the confidentiality of the file was secured.

## 6.3 Testing and Analysis on Steganography Technique

The aim for this testing was to verify the capability of the LSB algorithm to embed a modified sensitive object into a non-sensitive object (image carrier). The testing steganography procedure was derived from two main steps; verification of Steganography algorithm status, and verification of the percentage differences between an original image and stego-image.

### 6.3.1 Verification of status of Steganography Algorithm

This process verified the capability of the LSB algorithm to embed unintelligence information inside image carrier. Hence, the algorithm should be able to hide the sensitive data inside the image without obviously changing the difference between stego and original image. An output of the embedding process was a successfully created stego-image from the original image and sensitive data. Figure 11 shows the output of the steganography algorithm.

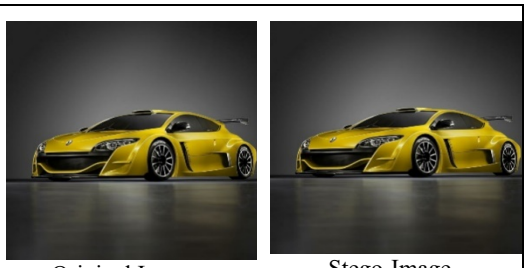


*Figure 11: Comparison Output of Steganography algorithm and the original image*

Based on Figure 11, the steganography algorithm was successfully executed as the algorithm was able to produce a stego-image from the original image with unintelligence information embedded in it. The stego image had no significant changes that could be identified using HVS. This shows that LSB algorithm technique was being implemented correctly and had produced the best result.

### 6.3.2 Verification of the percentage difference between original image and stego image

This testing phase had two sub processes namely the verification of the percentage differences in the image pixels between stego-image and original image and verification of the percentage increments in size of the stego-image from original image between two different LSB algorithms. Both testing phases were being carried out using the same image that was being resized using Adobe Photoshop according to the image dimension stated. This was to ensure the accuracy of the data collected as differences in density and complexity of RGB elements of image pixels would produce different results. The percentage of differences in image pixels between original image and stego-image conforming to the image dimension embedded with the same unintelligence information is shown in Table 4.

*Table 4: Difference in image pixels between stego image and original image*

| Image Dimension | Difference (%) |
|---|---|
| 500x500 | 0.3660 |
| 600x600 | 0.3482 |
| 700x700 | 0.3393 |
| 800x800 | 0.3334 |
| 900x900 | 0.3249 |
| 1000x1000 | 0.3249 |

Based on Table 4, an image with 500x500 dimension had 0.366% changes; an image with 600x600 dimension had 0.3482% changes; an

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

image with 700x700 dimension had 0.3393% changes; an image with 800x800 dimension had 0.3334%, changes; an image with 900x900 dimension had 0.3249% changes and an image with 1000x1000 dimension had 0.3249% changes between the stego image and the original image. The difference in image pixels between the original image and stego image is illustrated using line graphs as shown in Figure 12.

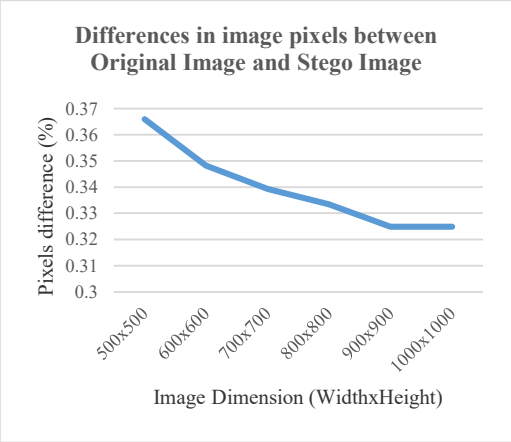


*Figure 12: Graph of differences in pixel between original image and stego image*

Based on Figure 12, the percentage of difference in image pixels between original image and stego image was decreasing as the image dimension increased. As LSB algorithm was implemented within this project, the differences between both images were logically small as the least significant bits was being manipulated, hence no big change occurred in the image pixels. Therefore, the graph indicated only slight changes which occurred between the original image and stego image which was less than 0.4%. Another test was done to verify the percentage increments in stego-image size from original image size between two different LSB algorithms.

This test focused on the differences between the stego-image and original image sizes. As there were additional data embedded into the original image, hence, the stego image size was logically increased. However, the steganography process that was being implemented in computer environment proved that the stego image size could remain the same even after the data were being embedded into the image. However, as the Trio-Security model had implemented the steganography technology in a mobile environment, the possibility of achieving the same results as in a computer environment was rather small as the image processing and manipulation library for mobile application development were not as advanced as image processing and manipulation library in desktop application development. The results of the increment of stego image size after undergoing the steganography process are shown in Table 5.

*Table 5: Increment of Stego-Image sized based on two different LSB algorithms*

| Image Size (KB) | Other LSB Algorithm Size Increment (%) | Trio-Security model LSB Algorithm Size Increment (%) |
|---|---|---|
| 42 | 178.6 | 103.6 |
| 43 | 265.0 | 134.5 |
| 45 | 304.4 | 139.2 |
| 52 | 313.5 | 140.4 |
| 55 | 345.5 | 129.1 |
| 61 | 365.6 | 109.8 |

Other LSB algorithms are the algorithms found on the internet in the Github directory whereas Trio-Security model LSB algorithm based is the LSB algorithm implemented in this model. Based on Table 5, the result showeds that the stego-image which used other LSB algorithms increased its size as the original image size increased. When stego-image used this LSB algorithm model, the size increased gradually but reduced after applying steganography process to 55KB and 61KB original image size. The graph in Figure 13 indicates the difference in an increment of stego-image size using two different LSB algorithms.

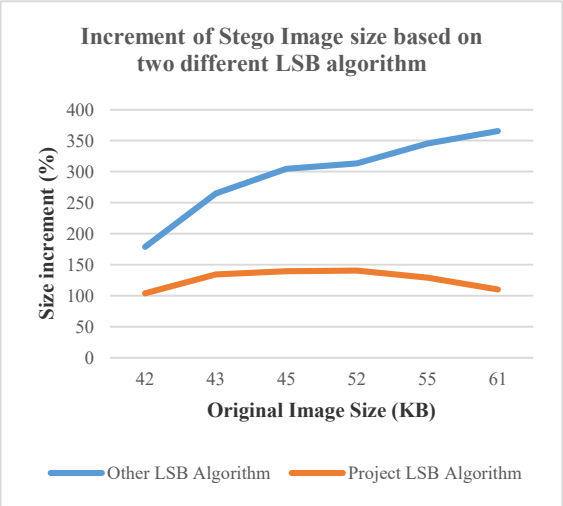


*Figure 13: Graph of increment in stego-image size using two different LSB algorithm*

Based on Figure 13, the stego-image size increased enormously using other LSB algorithm compared with using this project's LSB algorithm. As shown in the graph, using other LSB algorithms,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

the stego-image size increased up to a range of 180% to 360% of its original size. However, the increment of the stego-image size maintained below 150% when using this project algorithm for all the testing images. Yet, the size of the stego-image file was still increasing from its original image size compared with steganography technology implemented in computer environment.

This situation occurred to ensure sensitive data embedded inside the image were not corrupted or lost when reproducing the stego-image. Hence, the stego-image must be generated in PNG image format which was known as lossless image format. Therefore, no pixels of the stego-image was lost during the compression process to create the stego-image. This step was crucial to ensure the embedded data were not corrupted or lost as LSB algorithm technique produced stego-image that is easy to be manipulated. The compression into PNG image format had a bad effect on the stego-image as the PNG image size was increasing as the density and complexity in the depth of RGB color elements of the image increased. Hence, the result of this technique within the mobile environment generated a stego-image with high size increment. On different note, steganography within a computer environment was capable of generating a stego-image in JPG image format without suffering any pixel loss as the programming library contained ImageIO library. Hence, the stego-image could maintain its size equal to the original image size.

## 7. DISCUSSION AND FUTURE WORKS

This study, contribution is not limited certain group as it can be implemented in any environment, whether for individual, organization, company or university. This is because, most of the people in any environment will use data transfer application to transfer data between friends, colleagues and companies. Hence, by using this project, they can ensure that the data sent is secured from cyber attacks. As this study is developed for mobile environment makes a lot of people easy to access the application find it more convenient to use in order to send any data in a secure environment. Besides that, this study initiated file sharing for mobile environment that integrating three techniques (Trio-Security Model) which consists of four security measures, namely authentication, encryption, message digest and steganography in order to secure the data transmission. In future, a library of image processing and manipulation for mobile devices is needed to ensure the stego image produced using steganography algorithm provide the same quality whether in the computer environment or mobile environment.

## 8. CONCLUSIONS

The Trio-Security model helps to secure the confidentiality of data, provide the integrity of the data and ensure the real information will not available to the unauthorized user by concealing the presence of the data within other non-sensitive information. From this study, it enhances the data security while transmitted across the network by integrating the Cryptography technology, Message Digest technology and Steganography technology.

## REFRENCES:

[1] S. Malgaonkar, S. Surve, and T. Hirave, "*Distributed files sharing management: A file sharing application using distributed computing concepts*," In Proceeding of the IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), pp. 1-4, 2012.

[2] Gemalto NV, "It's All about Identity Theft", Technical Report, Breachlevelindex, 2016.

[3] Bajaj, S. B., & Grewal, M., "*TL-SMD: two layered secure message digest algorithm*", In Proceeding of IEEE International Advance Computing Conference (IACC), pp. 349-352, 2015.

[4] Verma, S. and Prajapati, G.S., "*Robustness and security enhancement of SHA with modified message digest and larger bit difference*", In Proceeding of IEEE Symposium on Colossal Data Analysis and Networking (CDAN), pp. 1-5, 2016.

[5] A. M. A. M. Asif and S. A. Hannan, "A review on classical and modern encryption techniques," *International Journal of Engineering Trends and Technology*, Seventh Sense Research Group, vol. 12, no. 4, pp. 199–203, 2014.

[6] G. Singh, "A Study of Encryption Algorithms (RSA, DES, 3DES and AES) for Information Security," *International Journal of Computing Applications*, Foundation of Computer Science, vol. 67, no. 19, pp. 975–8887, 2013.

[7] Kumar, A. and Pooja, K., "Steganography-A data hiding technique", *International Journal of Computer Applications*, Foundation of Computer Science, vol. 9, no. 7, pp.19-23, 2010.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[8] Akinola, S.O. and Olatidoye, "On The Image Quality And Encoding Times Of LSB, MSB and Combined LSB-MSB Steganography Algorithms Using Digital," *International Journal of Computer Science & Information Technology (IJCSIT)*, AIRCC Publishing Corporation, vol. 7, no. 4, pp. 79–91, 2015.

[9] Kaur, N. and Behal, S., "A Survey on various types of Steganography and Analysis of Hiding Techniques", *International Journal of Engineering Trends and Technology*, Seventh Sense Research Group, vol. 11, no. 8, pp.387-391, 2014.

[10] Bhanot, R. and Hans, R., "A review and comparative analysis of various encryption algorithms", *International Journal of Security and Its Applications*, SERSC Australia, vol. 9, no. 4, pp.289-306, 2015.

[11] Goodarzi, M.H., Zaeim, A. and Shahabi, A.S., "*Convergence between fuzzy logic and steganography for high payload data embedding and more security*", In Proceeding of the 6th IEEE International Telecommunication Systems, Services, and Applications (TSSA), pp. 130-138, 2011.

[12] Hamid, N., Yahya, A., Ahmad, R.B. and Al-Qershi, O.M., "Image steganography techniques: an overview", *International Journal of Computer Science and Security (IJCSS)*, CSC Journals, vol. 6, no. 3, pp.168-187, 2012.

[13] K. P. Singh and D. Kumar, "Performance Evaluation of Low Power MIPS Crypto Processor based on Cryptography Algorithms," *International Journal of Engineering Research and Applications (IJERA)*, vol. 2, no. 3, pp. 1625–1634, 2012.

[14] Christina, L. and Joe Irudayaraj, V.S., "Optimized Blowfish Encryption Technique", *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 2, pp. 5009-5015, 2014.

[15] A. K. Singh, "Steganography in Images Using LSB Technique", *International Journal of Latest Trends in Engineering and Technology (IJLTET)*, vol. 5, no. 1, pp. 426-430, 2015.

[16] Sawarkar, P.A., Nimkale, S.D. and Belsare, V.S., "Study and Comparison of Digital Image Steganography", *International Journal of Electronics, Communication and Soft Computing Science & Engineering (IJECSCSE)*, pp.270-275, 2015.