IT FOR SOCIETY, Vol. 04, No. 01
ISSN 2503-2224

Information Hiding Application Using Digital Images Secured With Rijndael and Caesar Cipher Encryption

Jonathan Surya[1]
Abdul Ghofir[2]
President University
Bekasi, Indonesia
[1]jonathan@gmail.com
[2]geoff@president.ac.id

Abstrak - Steganography is a way thtt conceals or embeds important data in form of text file into an image which possible to reduce and erase the existence of the data. A modern steganography can help millions of people to protect their data from potential threats or other individuals using computer programs and algorithm. This paper proposed an implementation of steganography Rijndael and Caesar schemes . This application will let the user embed the important data/information to an image and encrypt it.

1. Introduction

Steganography application is created to make the user's data safe and easy to use it. The application will be implemented in windows 10 operating system. By using this application, the user could prevent the percentage of several threats and increase their security needs. Cyber attacks in Indonesia are increasing year to year. That case has been attracting government to do more protection in order to save important things as many as possible, even though the security is already complex but someone will be able to bypass it.

Concerning those activities, it is better to hide the existence of the important data. in order to protect the data its better to eliminate the possible threat by hiding its existence. However this activities not only affected national level, it can find it in human daily life for example, when someone peeking other person phone while typing something in their phone or when someone able to bypass someone else password and peeking the phone text, media, etc.

The application works on windows platform and will act as a system which enable the user to conceal important data. The application is expected to be able to conceal or embed data inside the image without changing the image color significantly, in order to divert the potential attacker. Then the user can choose to add more security by using password(rijndael) and Caesar Cipher encryption, this also application has a user-friendly design which enabled the user to easy to use the application.

2. Digital Images

Digital images are made of picture elements called pixels. Typically, pixels are organized in an ordered rectangular array. The size of an image is determined by the dimensions of this pixel array. The image width is the number of columns, and the image height is the number of rows in the array. Thus the pixel array is a matrix of M columns x N rows. To refer to a specific pixel within the image matrix, we define its coordinate at x and y. The coordinate system of image matrices defines x as increasing from left to right and y as increasing from top to bottom Image file formats are standardized means of organizing and storing digital images.

Digital image files are composed of digital data in one of these formats that can be rasterized for use on a computer display or printer. An image file format may store data in uncompressed, compressed, or vector formats. Once rasterized, an image becomes a grid of pixels, each of which has a number of bits to designate its color equal to the color depth of the device displaying it. There are 4 possible images format that can be used in this application, such as: GIF, PNG, TIFF, BMP.

3. Rijndael Encryption

AES is based on a design principle known as a substitution–permutation network, and is efficient in both software and hardware. Unlike its predecessor DES, AES does not use a Feistel network. AES is a variant of Rijndael which has a fixed block size of 128 bits, and a key size of 128, 192, or 256 bits. By contrast, Rijndael *per se* is specified with block and key sizes that may be any multiple of 32 bits, with a minimum of 128 and a maximum of 256 bits. AES operates on a 4 × 4

column-major order array of bytes, termed the *state*. Most AES calculations are done in a particular finite field. The key size used for an AES cipher specifies the number of transformation rounds that convert the input, called the plaintext, into the final output, called the ciphertext. The number of rounds are as follows:

- 10 rounds for 128-bit keys.
- 12 rounds for 192-bit keys.
- 14 rounds for 256-bit keys.

Each round consists of several processing steps, including one that depends on the encryption key itself. A set of reverse rounds are applied to transform ciphertext back into the original plaintext using the same encryption key.

4. Steganography

Steganography is the art of concealing a file, message, image, or video within another file, message, image, or video. the first recorded uses of steganography can be traced back to 440 BC when Herodotus mentions two examples in his Histories. Histiaeus sent a message to his vassal, Aristagoras, by shaving the head of his most trusted servant, "marking" the message onto his scalp, then sending him on his way once his hair had regrown, with the instruction. detecting physical steganography requires careful physical examination, including the use of magnification, developer chemicals and ultraviolet light. It is a time-consuming process with obvious resource implications, even in countries that employ many people to spy on their fellow nationals. There are 3 types of steganography that can be group by its media, such as follows:

1 Image Steganography

Image Steganography uses image as its media to store important data into the colors that are generated from three primary colors as red, green and blue (RGB). various approaches has been designed for image steganography some of common approaches are LSB (Least Significant Bit) substitution which is the easy and most common approach of hiding data inside images.

2 Audio Steganography

Audio steganography uses soundwave files as its medium to store it's data. Audio steganography works by slightly changing the binary sequence and concealing with the secret message. Several methods are proposed such as Least Significant Bit (LSB) replacing last digit of carrier file. Parity coding involves breaking down of signal and then hiding the message in parity bits of each sample. Phase coding involves encoding of secret data to phase shifts.

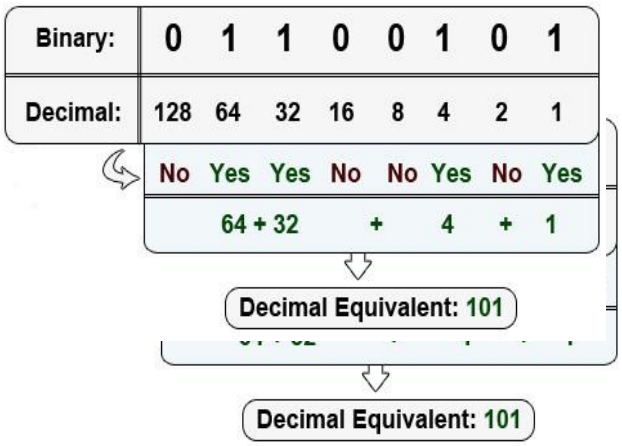
3 Video Steganography

Video Steganography uses video files as its medium to store data. The separation of video into audio and images or frames results in the efficient method for data hiding. The use of video files as a carrier medium for steganography is more eligible as compared to other techniques. As a result of this this technique is discussed and proposed in this paper.

5. Least Significant Bit Method

Steganography works by replacing bits of useless or unused data in regular computer files with bits of our important data. In our case, our data will be the plain text that we need to hide, and the unused data is the least significant bits (LSBs) in the image pixels.

Table 1 LSB table



Based on Table 1, it can be concluded first thing to do is to find the decimal value of a binary value, then sum all 1's in the binary value but multiplied by 2^n, where n is the position/index of the 1 from the right, starting from zero. For example, to convert 01100101 to decimal, then starting from the right, the equivalent decimal value is $1x2^0 + 1x2^2 + 1x2^5 + 1x2^6 = 1 + 4 + 32 + 64 = 101$

The least significant bit (LSB) is the bit that when flipped from 0 to 1 or from 1 to 0, then no significant change will occur on the total value. It's the bit on the rightmost, that when flipped, the value will be only affected by 1 to be 100 instead of 101. This means that the image will not be significantly affected when we reserve this bit for our

purpose. Where the most significant bit (MSB) is bit on the leftmost, that when flipped, the value will be affected by 128 ($1x2^7$) to be 229 instead of 101.

When the application tries to hide the user data in an image, then the application needs enough budget of LSBs to hide the user data in. These bits are located in the image pixels. Since each pixel has three elements (R, G, and B that represent the Red, Green, and Blue elements of the pixel consecutively, assuming non-transparent image), each of these elements can have a value between 0 and 255. Now, assume that the image was 300 pixels width by 400 pixels height, then we'll have 300 x 400 x 3 = 360000 LSBs. And as each character can be represented by 8 bits, then that image can hide 360000 / 8 = 45000 characters.

6. Limitation of The Application

This application only available using image as the media to store text file data, the main reason steganography was created to hide the existence of the data in order no human would steal or attacked it. This application is created based on that principle, although this application has password protected featured and Caesar Cipher encryption it is only serve as weak defense layers to current system. Also the Least Significant Bit has a maximum amount limit by using this formula.

The maximum error possible using the technique of last 4 bits steganography is ±15 in the values of each component RGB. The maximum error percentage possible in each component value is 5.88%. Since, this is the maximum error possible, if the 4th last bit remains unchanged, the error will reduce to ±7 and if the 3rd last bit also remains unchanged, the error will further reduce to ±3. This is likely to be imperceptible or faintly perceptible by the human eye. Image in Image: We consider a carrier image of M × N resolution and the message image of P × Q resolution with the same aspect ratio. The numbers of bits in the message image are 'B'. The maximum numbers of bits that can be encrypted using the 4-bit steganography in the carrier image are 'C'.

B= Resolution of Image × Number of Colour Components × Bit Depth of each Component (1)

C= Resolution of Image × Number of Colour Components × 4 (2)

Example:

For a High Definition 'message' bitmap,

B= 1280 * 720 * 3 * 8

B= 22118400

For a Full High Definition 'carrier' bitmap,

C= 1920 * 1080 * 3 * 4

C= 24883200

Since the value of C is greater than the value of B, the 'message' image can be encrypted.

7. Test Cases

The test cases chapter explains the testing of the program to evaluate all the features with its specified requirements. The system testing is performed to identify the application will run properly and can achieve the expected result. There are three sections that will be described in this section.

The feature test will include several users to test the user experience of the windows application. Some random images are taken from internet to test the application, the users also will be asked to operate some of task given by the instructor and the users can freely to try every feature in the program.

The main purpose of this testing scenario is to ensure the application runs well and also to organize the test more systematically. Not only that testing scenario has a several risk such as When the product is unstable, scenario testing becomes complicated, Scenario testing are not designed for test coverage.

Scenario tests are often heavily documented and used time and again. If the application passed every test then the application is ready to be deploy or shared to a group of people in order to simplify their security needs. The testing case cases for every features and result can be shown in the next page Table 2 Functionality testing scenario.

8. Conclusion

There are few points that can be concluded. There are two points concluded from the development of this application. The Least Significant Bit Method is successfully implemented in Steganography application, which is capable to embed the secret message into an image. Lastly, the rijndael algorithm and caesar cipher algorithm is successfully implemented as one of the features of this application which is capable to encrypt the secret message stackable by each algorithm.

In this paper image based steganography methods have been proposed to reduce the existence of the data and encrypt it using Rijndael and Caesar cipher. It focuses on the impact

and development of image steganography application that can hide data with a low detection rate and secure with encryption.

It can be concluded that the Least Significant Bit, Rijndael, and Caesar Cipher Algorithms can be used to secure data or information being exchanged because the application fulfils three things of data security which are robustness, capacity, and security.

Table 2 Functionality testing scenario

No.	Testing Page	Scenario	Expected Result	Result
1.	Upload an Image	Click upload button	Window of open file will open.	Passed
		Choose one image and click open.	Image will be displayed in picture box	Passed
2.	Save an Image	Click save button.	Window of save file will open.	Passed
		Choose a save directory and click save.	A message box will be pop up to notify the user	Passed
4.	Upload a text file	Click upload button.	Window of open file will open.	Passed
		Choose one text file and click open.	The text will be displayed in text box.	Passed
5.	Save a text file	Click save button.	Window of save file will open.	Passed
		Choose a save directory and click save.	A message box will be pop up to notify the user	Passed
6.	Embed a message	Upload an image.	Image will be displayed in picture box.	Passed
		Input secret message or load a text file.	The text will be displayed in the textbox.	Passed
		Click embed button.	A label will be change into “don’t forget to save the image”.	Passed
		Save the image by clicking save button.	Window of save file will open.	Passed
		Choose the directory	A message box will be pop up to notify the user	Passed
7.	Extract a message	Upload a stego image.	Image will be displayed in picture box.	Passed

		input password and encryption key(if needed) and click extract button.	The secret message will be displayed in the textbox.	Passed
8.	Apply Password	Check password checkbox.	The password text box will be enabled.	Passed
		Input password in textbox and click apply.	An alert message will be displayed.	Passed
9.	Apply encryption	Check encryption checkbox.	The encryption text box will be enabled.	Passed
		Input encryption in textbox and click apply.	An alert message will be displayed.	Passed

References

[1] Patrick Philippe Meier. “Steganography 2.0: Digital Resistance Against Repressive Regimes”, Retrieved 17 June 2010, http://www.irevolution.wordpress.com.

[2] Mazurczyk, Wojciech; Wendzel, Steffen; Zander, Sebastian; Houmansadr, Amir; Szczypiorski, Krzysztof. “Information Hiding in Communication Networks: Fundamentals, Mechanism, and Applications”, Retrieved Febuary,2016, ISBN 978-1-118-86169-1.

[3] John Kensley, Stefan Lucks, Bruce Schneier, Mike Stay, David Wagner, and Doug Whiting.”Academic: Improved Cryptanalysis of Rijndael” Retrieved 2007-03-06. https://www.schneier.com/academic/archives/2001/01/improved_cryptanalys. html.

[4] Reinke, Edgar C. "Classical Cryptography", Retrieved December, 1992, The Classical Journal. 58 (3): 114.

[5] Goldreich, Oded. Foundations of Cryptography: Volume 2, Basic Applications. Vol. 2. Cambridge university press, 2004.

[6] Leighton, Albert C. "Secret Communication among the Greeks and Romans". Technology and Culture. Retrieved April 1969,10 (2): 139–154.

[7] Luciano, Dennis; Gordon Prichett (January 1987). "Cryptology: From Caesar Ciphers to Public-Key Cryptosystems". The College Mathematics Journal. 18 (1): 2–17. 91

[8] Digital Negative (DNG) Specification. San Jose: Adobe, 2005. Vers. 1.1.0.0. p. 9. Accessed on October 10, 2007.

[9] Joan Daemen and Vincent Rijmen (September 3, 1999). "AES Proposal: Rijndael"(PDF). Archived from the original (PDF) on February 3, 2007.

[10] Daemen, Joan; Rijmen, Vincent (March 9, 2003). "AES Proposal: Rijndael" (PDF). National Institute of Standards and Technology. p. 1. Archived (PDF) from the original on 5 March 2013. Retrieved 21 February 2013.

[11] Kamaldeep joshi (25 December 2017). "A New Method of Image Steganography Using 7th Bit of a Pixel as Indicator", Published 1 August 2018.