

Analysis and Utilization of the Base64 Algorithm for Image Encryption and Decryption Security in Web-Based Images

Fadhilrahman Baso^{1*}

Pend. Teknik Informatika dan Komputer
Universitas Negeri Makassar
Makassar, Indonesia
fadhilrahman.baso@unm.ac.id

ARTICLE INFO

Received : 31 October 2023
Accepted : 27 November 2023
Published : 09 December 2023

ABSTRACT

Data is often the target of irresponsible people for misuse. The theft is done to benefit from the person who has the data. In addition to theft of work files, theft is also carried out on image files. The purpose of this file theft is to find out what the image contains. Someone has a private image that cannot be known by others. Misuse of image files will be fatal for the owner of the image. Cryptographic techniques are needed in securing images and one of the methods used to secure these images is using the base64 algorithm. The Caesar Cipher algorithm can help users secure the image file. The Base64 algorithm can be used to change the ASCII 256 format to Base64 so that it is easy to send or store on a storage medium. By applying the Base64 algorithm to image files, the security and confidentiality of these files will protect data from irresponsible people.

Keywords: Data Security, Base64, Encryption, Decryption

ABSTRAK

Data sering kali menjadi sasaran orang yang tidak bertanggung jawab untuk disalahgunakan. Pencurian yang dilakukan adalah untuk mendapatkan keuntungan dari orang yang memiliki data tersebut. Selain pencurian file-file kerja, pencurian juga dilakukan terhadap file gambar. Pencurian file ini bertujuan untuk mengetahui apa isi gambar tersebut. Seseorang memiliki gambar pribadi yang tidak boleh diketahui oleh orang lain. Penyalahgunaan file gambar akan berakibat fatal bagi pemilik gambar tersebut. Teknik kriptografi diperlukan dalam pengamanan gambar dan salah satu metode yang digunakan untuk mengamankan gambar tersebut adalah menggunakan algoritma base64. Pada algoritma Caesar Cipher dapat membantu pengguna dalam mengamankan file gambar tersebut. Algoritma Base64 dapat digunakan untuk mengganti format ASCII 256 menjadi Base64 sehingga mudah untuk dikirimkan atau disimpan dalam suatu media penyimpanan. Dengan menerapkan algoritma Base64 pada file gambar, keamanan dan kerahasiaan file tersebut akan melindungi data dari orang yang tidak bertanggung jawab.

Kata Kunci: Keamanan Data, Base64, Enkripsi, Dekripsi

This is an open access article under the CC BY-SA license



I. INTRODUCTION

The development of data and information and communication technology is currently an important part of everyday human life. As time goes by, human need for information is increasing. security is a state of being free from danger[1]. Security can be implemented into various things, including data and information. Data is an important asset in the survival of companies, government agencies, educational institutions and even individuals. Security issues are one important aspect of an information system. One of the important things in communication using computers to ensure the security of messages, data or information is encryption. Here encryption can be interpreted as a code or chipper[2].

A website is a means used to disseminate information via the Internet, in the form of text, images, sound or video. As the use of websites becomes more widespread, it can lead to various kinds of crimes such as theft, manipulation of important information data from a website by irresponsible people [3].

Encryption is a process where information or data to be sent is changed into a form that is almost unrecognizable as the original information using certain algorithms [4]. Furthermore, in carrying out web-based image security, a coding system uses a table or dictionary that has been defined for the word information or which is part of the message, data or information sent [5]. Then the chipper uses an algorithm that can encode all data streams (streams) of bits from an original message (plaintext) into an incomprehensible cryptogram [6]. Because the chipper system is a system that is ready to be automated, this technique is used in web-based system security. One of the data that is often manipulated and misused is digital images. Manipulated images can damage someone's good name[7]. This of course violates everyone's right to privacy. Thus, efforts to secure digital data become very important and very urgent.

The goal is to explain in detail how the Base64 algorithm works, including the encryption and decryption process. Focuses on how Base64 converts binary data to text so that it can be integrated in the context of image security. Understanding the Importance of Data and Information Security. Highlighting how important data and information security is in this modern technological era to protect vital assets, both for companies, government institutions, educational institutions and individuals. The Role of Encryption in Protecting Data and Information[8]. Explores the role of encryption techniques, including algorithms such as Caesar Cipher

and Base64, in protecting messages, data, and information sent via digital communications. Focus on how encryption can maintain the confidentiality and integrity of information.

II. METHODS

A. Research Flow

In this research, the method used is experimental, meaning that the research is carried out to carry out trials on certain problems using certain theories so that appropriate test results are obtained between the problems taken and the theory used. Theories that can be used for test results include:

1. Needs Analysis, In the context of using the Base64 algorithm on web-based images, the need must consider whether the Base64 algorithm will be used to encrypt images or only used to reduce image size [9]. It will then consider the type of image data used and the types of security attacks that may occur on the image, such as XSS (Cross-Site Scripting) attacks or SQL injection attacks. By considering these factors, the next step is to help develop appropriate solutions for web-based image security using the Base64 algorithm[10].

2. Building the System, In this stage, a design of the system architecture that will be used is created. The system architecture should include Base64 Encryption and Decryption algorithms to secure images and features to ensure data security.

3. Product Testing: The system must be tested to ensure that the system works well and is safe. The testing phase includes system testing using various test cases and simulations. After successful testing, the system must be maintained periodically to ensure that the system remains secure and functions properly.

B. System Design

This system consists of two main components: image encryption using the Base64 algorithm, and image decryption using the Base64 algorithm. The process design used by the system that has been designed to carry out the encryption and decryption process using the Base64 algorithm is as follows.

In this flowchart there are two processes, namely Encryption and Decryption, where in the Encryption process section enters a document or image data that has not been previously encrypted. The image that is input is an image that matches the previously determined format. Then in processing, if the image does not match the format, it will return to the data input section, but if the data format is appropriate, it will produce encrypted text.

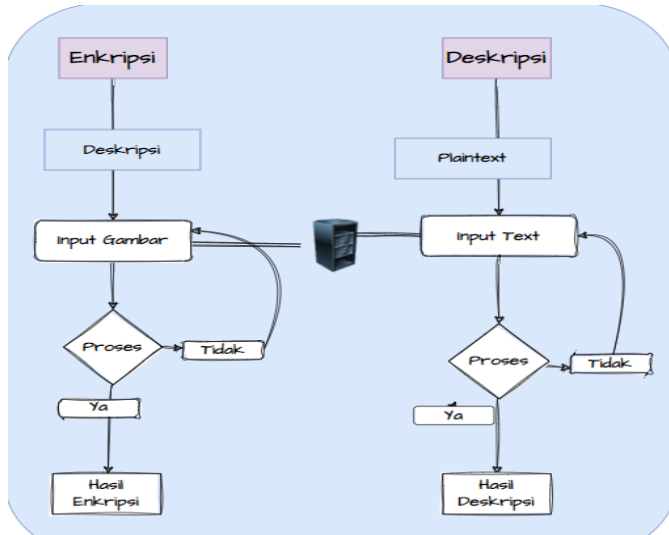


Fig. 1. Decryption Encryption Process Flowchart

In the description section, the user will enter the text resulting from the encryption, if it matches the text, it will produce data that can be understood by humans, if it is not appropriate, it will return to the text input section.

C. System Architecture

The following is the architecture of a web-based image security system using the base 64 algorithm as follows:

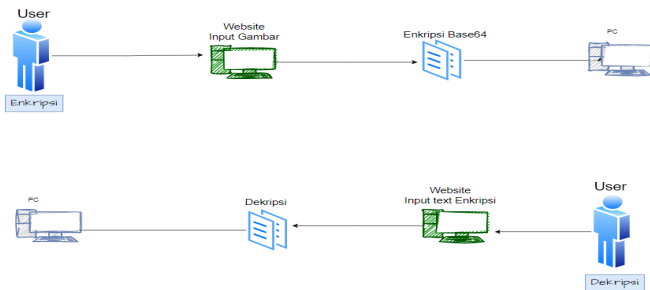


Fig. 2. System Architecture

1) Encryption

- User accesses it
- Then input the image based on the image type that has been determined
- Next it will be processed by the algorithm that we have determined, namely Base64, and
- The image encryption results will be displayed on the PC in text form that is difficult for humans to understand.

2) Decryption

- User accesses it.

- Then input the encryption result text based on what has been determined.
- Next, it will be processed by the system to restore the original image, and
- The image decryption results will be displayed on the PC in the form of an image that humans can understand.

D. Base64 Algorithm

In this system, the Base64 algorithm will be applied which will be used to secure image files. The psoudecode used in this system is based on the base64 algorithm in encrypting images as follows:

```
Deskripsi
Read ("jpg""jpeg""png""gif""bmp")
if ("gambar""jpg""jpeg""png""gif""bmp")
    $POST_Base64
else
    $valid_array
if POST
Endif
```

Fig. 3. Encryption

The code provided uses the PHP programming language to upload images and perform base64 encryption on the image files. Here is a brief explanation of the code:

- First, define an array containing valid image file extensions (jpg, jpeg, png, gif, bmp).
- Then, check whether the submit button has been pressed and the image file size is more than 0.
- If the check is successful, extract the image file extension using the explode () function to separate the file name from the extension, then the strtolower () function is used to change the extension to lower case.
- Next, check whether the extracted image extensions are contained in the valid extension array that has been previously defined using the in_array () function. If the extension is valid, then the next step is to move the uploaded image file to the "images" directory by using the move_uploaded_file() function.

After that, base64 encryption is carried out on the uploaded image files using the base64_encode() function and the file_get_contents() function. The encryption results are stored in the \$encryption variable.

E. System Interface

Design a user interface that allows users to upload images to encrypt or encrypted images to decrypt.

Include a form where users can select an image file and a button to start the encryption or decryption process. Show an error message if a problem occurs during the encryption or decryption process.

In this display is the Image Upload Form. This section contains a form that allows users to select and upload image files that will be encrypted. By using the `<input type="file">` element for this. Submit Button: This button is used to submit the form and start the image encryption process. This display is a simple display that displays a form for uploading images and a submit button. Once the user selects and uploads an image, the encryption process will begin.

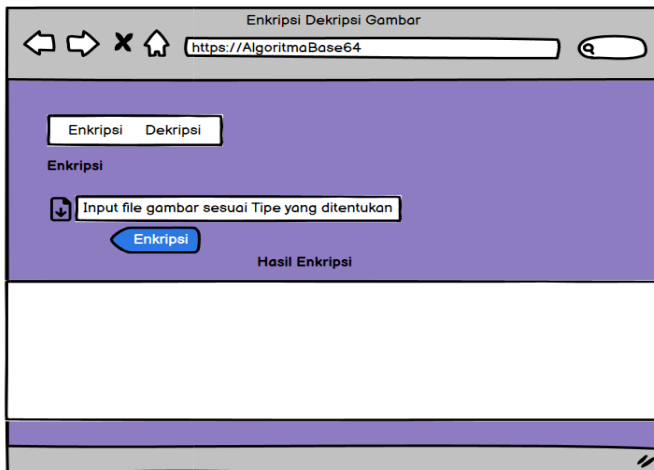


Fig. 4. Encryption Appearance

1. This display is the Encrypted Image Upload Form: This section contains possible forms.

2. The user selects and uploads the encrypted image file to be decrypted. You can use the `<input type="file">` element for this.

3. Submit button: This button is used to submit the form and start the image decryption process. This display is a simple display that displays a form for uploading an encrypted image and a submit button. Once the user selects and uploads an encrypted image, the decryption process will begin.

4. This display is the Encrypted Image Upload Form: This section contains a form that allows users to select and upload an encrypted image file that will be decrypted. You can use the `<input type="file">` element for this.

5. Submit button: This button is used to submit the form and start the image decryption process. This display is a simple display that displays a form for uploading an encrypted image and a submit button. Once the user

selects and uploads an encrypted image, the decryption process will begin.

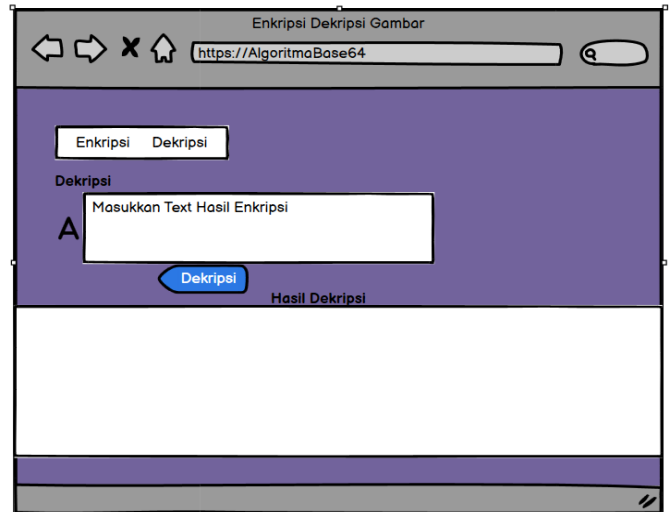


Fig. 5. Decryption Appearance

III. RESULT AND DISCUSSION

A. System Needs

Chrome is a popular choice for web application testing due to its ability to perform automated testing, support for a wide range of devices and resolutions, and good integration with web development tools. By using the Chrome Testing Environment, developers can improve the quality and reliability of their web applications through comprehensive and efficient testing. Here is software and hardware specification:

- Windows 10 pro64-bit
- Laptop
- Intel® Core™ i3-500U CPU@2.00GHz (4CPUs)-2.0GHz
- 4096MB GB of RAM

B. Determine the Image

The process for selecting an image file that will be converted into a text representation in base64 form. The following is an explanation of how to determine which images will be encrypted:

Input from user:

You can allow users to upload images and use files uploaded by users as images to be encrypted. Generally, this involves using the `<input type="file">` element in an HTML form that allows users to select an image file from their device. Once an image is uploaded by a user, you can access the file via the `$_FILES` variable in PHP. The `$_FILES` variable stores information about the uploaded file, including the temporary path on the server where

the file is stored. You can use the path to read the contents of the image file, apply base64 encryption, and save the result.

In the process of determining the image, the image to be input must have an image format that is supported for encrypting, which is as follows:


1. .jpg
2. .jpeg
3. .png
4. .bmp
5. .gif
6. .psd

By encrypting images using the Base64 algorithm in the encode and decode process method, image security will be safer because not just anyone can see the image.

C. Image Testing

After determining the image that we have determined which can be used as testing material based on the format specified in the system that can be base64 encrypted, namely the format: jpg.jpeg.png.bmp.gif.psd.

Table 1. Encryption Image Testing

Input	Ouput
	<pre>/9j/4AAQSkZJRgABAQEBA AAAAAAIAAAITAA</pre>

In the image above is an image test on a system that has been previously designed, so here we will test the image by encrypting it using that system.


1. First we will open the system
2. Then select the menu section, namely encryption, because here we will test encryption on images using a system that uses the base64 algorithm
3. Next, enter the image in the input file, the input image file must match the image format that has been specified, namely: jpg.jpeg.png.bmp.gif.psd, if the input form is not detected, the format in the image is not detected. This means that the image entered is an error or does not match the format that has been determined. However, if the image format on the form is detected, the image is in accordance with the format specified previously.
4. Then click on the submit or encryption button and it is processed by the system.

After completing image testing in the input form in the encryption section, it will be submitted to

produce encryption results for the image which has been processed by the base64 algorithm.

Then, after completing the encryption, here we will again decrypt the image encryption results, where the encryption results cannot be understood by humans, so they can be understood by humans. The decryption testing is:

Table 2. Decryption Image Testing

Input	Ouput
<pre>/9j/4AAQSkZJRgABAQEBA AAAAAAIAAAITAA</pre>	

1. First we open the system then select the decryption menu.
2. Then enter the encryption results for the image in the text input form (Plaintext).
3. After that, click on the submit or decrypt button to see the decryption results on the previously encrypted image.

IV. CONCLUSION

The Base64 algorithm is not designed for strong encryption security. The main purpose of Base64 is to convert binary data into a text format that can be transmitted over protocols that do not support binary data. Encryption and decryption using Base64 is not a secure encryption method. Base64 simply hides the original data by converting it to a different text format. Information encrypted using Base64 can be easily converted back to its original form by anyone with knowledge of this method. Base64 is more often used for simple encryption purposes, such as hiding non-sensitive information or used in simple authentication processes. Examples of use are when sending images in the form of URL data in web applications, or when sending data that requires special text representation, such as authentication tokens. Although Base64 is not suitable for secure encryption, if you want to improve the security of images on the web, there are stronger encryption methods to consider, such as using the HTTPS protocol to protect image transfer between the server and the user, or using secure symmetric or asymmetric encryption algorithms to secure sensitive image data.

It is important to remember that security is a very important aspect in web-based systems, and it is important to consider the use of encryption algorithms

appropriate to the level of security required in cases of image data transmission.

REFERENCES

- [1] H. Agusta, "Keamanan dan Akses Data Pribadi Penerima Pinjaman Dalam Peer to Peer Lending di Indonesia," vol. 15, no. 1, pp. 11–38, 2021.
- [2] B. Untuk and M. Email, "Penerapan algoritma kriptografi tea dan base64 untuk mengamankan email," vol. 2, no. 1, pp. 34–41, 2018.
- [3] R. W. Abdullah, S. Wulandari, and F. P. Nugroho, "Keamanan Basis Data pada Perancangan Sistem Kepakaran Prestasi SMAN dikota Surakarta," vol. 12, no. 1, pp. 13–21, 1978.
- [4] R. R. Rachmawati, "Smart Farming 4.0 Untuk Mewujudkan Pertanian Indonesia Maju, Mandiri, Dan Modern," *Forum Penelit. Agro Ekon.*, vol. 38, no. 2, p. 137, 2021, doi: 10.21082/fae.v38n2.2020.137-154.
- [5] R. I. Adam, "Skema Penyembunyian Data pada Gambar Berbasis Interpolasi Kubik B-Spline Menggunakan Metode Least Significant Bit (LSB)," vol. 5, no. 3, pp. 255–260, 2019.
- [6] C. Lika *et al.*, "Jurnal Computer Science and Information Technology (CoSciTech)," vol. 4, no. 1, pp. 200–206, 2023.
- [7] N. I. Putri, R. Komalasari, and Z. Munawar, "Pentingnya Keamanan Data Dalam Intelijen Bisnis," *J. Sist. Inf.*, vol. 1, no. 2, pp. 41–49, 2020.
- [8] H. P. Yuwinanto, "Privasi online dan keamanan data," no. 031.
- [9] S.- Sallu and Q. Qammaddin, "Keamanan Data Pembelajaran Online Jaringan Komputer Di Perguruan Tinggi," *Instruksional*, vol. 2, no. 1, p. 35, 2020, doi: 10.24853/instruksional.2.1.35-40.
- [10] D. A. Wp, "Peningkatan Keamanan Data dengan Metode Cropping Selection Pseudorandom," vol. 4, no. 3, pp. 132–138, 2016.