ISSN 2087-3336 (Print) | 2721-4729 (Online)

TEKNOSAINS: Jurnal Sains, Teknologi dan Informatika
Vol. 12, No. 1, 2025, page. 104-109
http://jurnal.sttmcileungsi.ac.id/index.php/tekno
DOI: 10.37373

Cryptographic security analysis of the shift password method using the Google colab application

Rokhim Utomo*, Mokhammad Syafaat, Kasiyanto, Dekki Widiatmoko, Rafi Maulana

* Politehnik Angkatan Darat, Indonesia, 65311

* rohimmadiun@gmail.com

Submitted: 19/06/2024 Revised: 14/07/2024 Accepted: 09/09/2024

ABSTRACT

The study of information security methods to prevent unauthorized parties from reading it is known as cryptography. The Caesar Cipher method is a low-tech yet well-established cryptography technique. One of the earliest and most basic cryptographic methods is the Caesar Cipher, which substitutes a different letter with a specific alphabetic difference for each letter in the text. The Caesar Cipher is simple to comprehend and use, but it has a lot of flaws that leave it open to frequency analysis and brute force attacks. The purpose of this study is to use the Google Colab tool to examine the cryptographic security of the Caesar Cipher technique. In this study, Google Colab—a platform that offers a robust and user-friendly Python programming environment—was utilized to construct and evaluate the Caesar Cipher algorithm. This study design combines an experimental strategy with a descriptive method. The Caesar Cipher technique is put into practice, encryption and decryption simulations are run, and character frequency analysis and brute force assaults are used to analyze security flaws. The study's findings demonstrate that the character distribution pattern is preserved when character frequencies in the original and encrypted text are seen, making it simple for attackers to use frequency analysis to crack the password. This suggests that, in the context of high information security, the Caesar Cipher should not be used. Shift passwords are therefore inappropriate for use in contemporary security applications due to their serious flaws.

Keywords: Cryptography; shift cipher; Google colab; encryption; decryption.

1. INTRODUCTION

Cryptography is a strong and efficient way to protect the privacy of data. Cryptographic systems use intricate mathematical procedures that frequently need large computational resources to secure information [1]. Shift cipher is one of the oldest and most basic encryption techniques. Even with its ease of use, this method's security analysis is necessary to comprehend the fundamentals of cryptography. In shift cipher cryptography, every letter in the original text is changed to a different letter and shifted to a predetermined number of steps in the alphabet. This sort of substitution cipher is known as a monoalphabetic cipher [2]. By using this method, the desired communication is encrypted such that only specific individuals with the appropriate key can decrypt it [3]. Grasp the benefits and drawbacks of this cryptographic method requires a grasp of the shift cipher's security [4]. The message signature and key length are two factors that can impact the shift cipher's level of security [5].

Google Colab is a cloud-based platform that lets you write, execute, and share Python code via a web browser without requiring any device installations [6]. Google Colab makes it possible to analyze, fast and effectively, Python code that is running and shared over a web browser without requiring shift cipher encryption. Researchers may evaluate security strength, use encryption and decryption techniques, and find potential shift password flaws with Google Colab capabilities [7]. Thus, the platform utilized in this study to model and assess the security of shift passwords is Google Colab. Prior research has addressed more complex encryption techniques and real-world applications, highlighting the significance of robust security solutions that advance the field of cryptography by addressing a range

of issues and complexity levels related to data security and encryption. Prior research has employed techniques such as the Data Encryption Standard (DES), the Reverse Cipher, and the RSA approaches to secure guest data at hotels [8][9]. This paper explains shift passwords and their vulnerabilities and demonstrates how to do cryptography research with Google Colab [10]. Both studies address data security and encryption; the distinction is that the current study focuses on ease of simulation and analysis and uses Google Colab for implementation, whereas earlier studies used RSA and DES to demonstrate their application for secure data transmission and storage.

The purpose of this study is to evaluate the shift cipher method's security flaws by utilizing Google Colab as a platform for development and simulation [11]. Because Google Colab offers a robust and user-friendly Python programming environment, it was selected to enable researchers to carry out experiments effectively [12]. It is anticipated that this study will deepen our understanding of shift ciphers and highlight Google Colab's potential as a resource for cryptography research.

2. METHOD

An experimental descriptive method is used in this investigation. The platform for implementation and simulation was Google Colab since it utilizes robust and user-friendly Python programming [13]. One of the study phases is to write code for text encryption and decryption in Google Colab by developing a shift cipher algorithm. To test the encryption and decryption methods with a shift key value, a simulation will be run in the following phase. Understanding how the shift cipher functions and how the original message can be encoded and transformed back into its original form is helped by this step. Testing the algorithm for possible attacks allows security analysts to assess its vulnerabilities. Brute force attacks are particularly effective because they may be used to try every key—from 1 to 26—to decrypt encrypted material until the right key is found [14].

Additionally, to estimate the key used, researchers performed a frequency analysis by examining the frequency of character appearances in encrypted text and comparing it with the character frequency. Since the letter "E" is the most frequently occurring in English, for instance, its predominance in encrypted text may suggest the presence of a shift key. The analysis's findings will offer comprehensive details regarding the system under development. The process that will be utilized to create the system architecture is as follows:

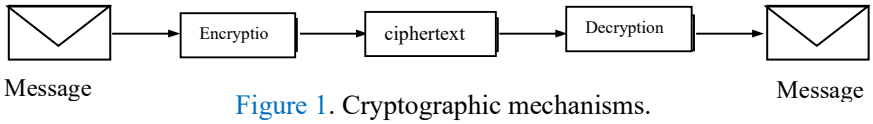

Figure 1. Cryptographic mechanisms.

As can be seen in Figure 1 the procedure is essentially extremely straightforward [15]. The encryption procedure will transform a message (plaintext) into a ciphertext. The message will be produced again by the ciphertext through the description process to retrieve it [16].

Documentation and interpretation of results: The last stage is recording and interpreting the results of the implementation and analysis that have been done. Based on the results of frequency analysis and brute force attacks, the efficacy of the shift code will be assessed in the interpretation of the data.

3. RESULTS AND DISCUSSION

The outcomes of the shift password algorithm's implementation, simulations of encryption and decryption, and an examination of security flaws discovered throughout the investigation will all be explained by this analysis. The shift password algorithm's application. Google Colab implements the shift password algorithm in the first phase. The code used for encryption and description is as follows.

Figure 2. Text encryption algorithm.

The encryption process algorithm is shown in Figure 2. Each character in the plaintext is shifted by a predetermined number of positions (shifts) in the alphabet using this algorithm. 'A' becomes 'D' and 'B' becomes 'E', for instance, if the shift is 3.

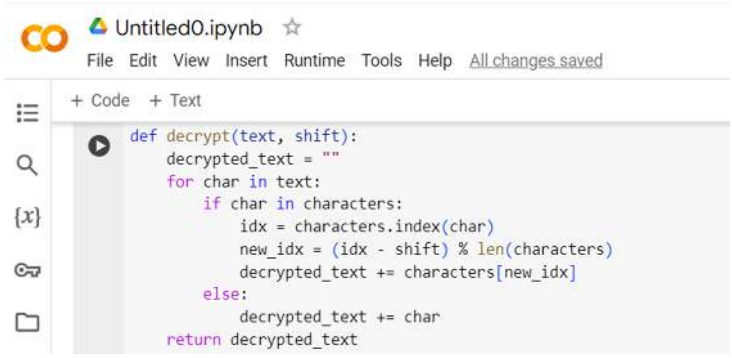

Figure 3. Text decryption algorithm.

A decryption technique that restores each character in the ciphertext to its original alphabetic place is shown in Figure 3. Testing of encryption and decryption. To test, plaintext input is shifted, then the text is encrypted and decrypted.

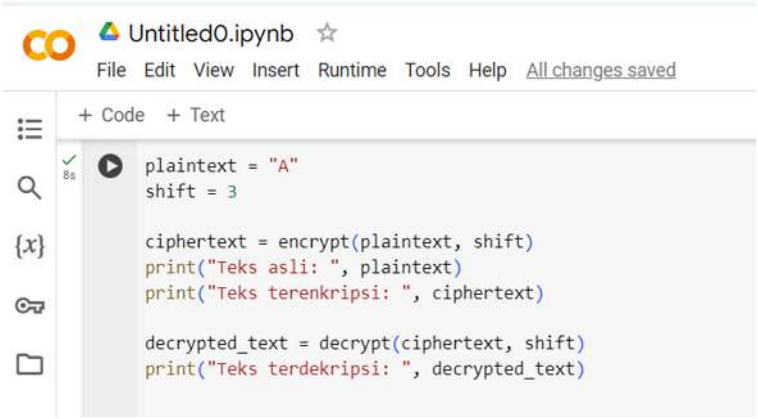

Figure 4. Examples of algorithms for testing encryption and decryption.

To ascertain whether the encryption and decryption process was successful, the researchers carried out various observations using various shift bits and text lengths. Table 1 contains a list of the observations' outcomes.

Table 1. Encryption and decryption test results.

Slide Input	Plaintext (Encryption)	Chipertext	Plaintext (Description)
1	A	B	A
2	Elka	Hond	Elka
4	Tehnik	Xilrmn	Tehnik

Table 1 demonstrates that the ciphertext may be found on the shift input, where one shift bit was used for the first test, two shifts for the second test, and four shifts for the third test. By contrasting the ciphertext's decoding results with the original plaintext, it is possible to determine the encryption and decryption test results table's success. The encryption and decryption procedure is deemed successful if the decrypted text is identical to the original plaintext.

The following is a security flaw study of Google Colab's shift password cryptography implementation.

a. Brute force attack

Since there are only 26 alphabets in shift ciphers, one of their key disadvantages is that they are simple to break using brute force attacks. An attacker can try every shift until they locate the original text.

```
def brute_force_decrypt(ciphertext):
    for shift in range(26):
        decrypted_text = decrypt(ciphertext, shift)
        print(f"Shift {shift}: {decrypted_text}")

ciphertext = "HOND"  # hasil enkripsi dari "ELKA" dengan shift 2
brute_force_decrypt(ciphertext)

Shift 0: HOND
Shift 1: GNMC
Shift 2: HOND # teks asli ditemukan
```

Figure 5. Example of a brute force attack.

Figure 5 shows that a brute force attack can be carried out by searching for each shift one by one until the original text is found.

b. Frequency analysis

The distribution of characters in the ciphertext, which is consistent with the plaintext, is another flaw. An attacker can determine the shift utilized by looking at how frequently certain characters appear in the ciphertext. To determine the shift that was employed, one might examine the ciphertext's character distribution pattern.

```
import matplotlib.pyplot as plt

def plot_frequency(text, title):
    frequency = {}
    for char in text:
        if char in frequency:
            frequency[char] += 1
        else:
            frequency[char] = 1
    plt.bar(frequency.keys(), frequency.values())
    plt.xlabel('Karakter')
    plt.ylabel('Frekuensi')
    plt.title(title)
    plt.show()

plaintext = "ELKA"
ciphertext = encrypt(plaintext, 2)
plot_frequency(plaintext, "Frekuensi Karakter dalam Plaintext")
plot_frequency(ciphertext, "Frekuensi Karakter dalam Ciphertext")
```

Figure 6. Frequency analysis in Google Colab.

If you enter a code like Figure 6, the character frequency in plaintext and ciphertext will appear as in Figure 7.

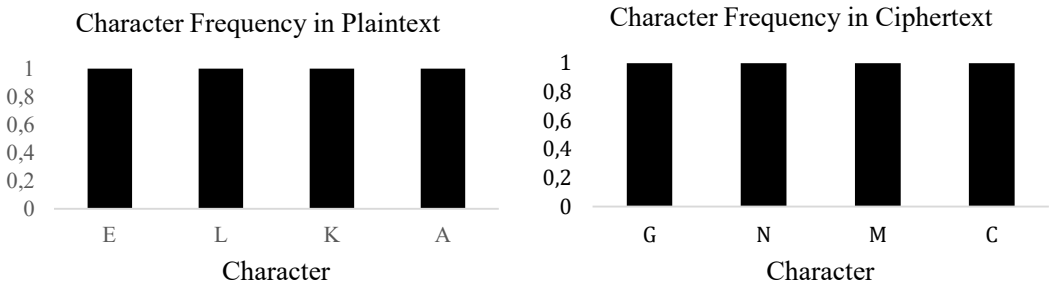

Figure 7. Character frequency results in plaintext and ciphertext.

The almost same character distribution pattern between the ciphertext and the plaintext is seen in Figure 7, which an attacker could use to crack the shift cipher. Even though the characters have been encrypted, the frequency pattern of the characters in the plaintext and ciphertext can still be determined. This demonstrates that one of the shift cipher's vulnerabilities is that adversaries can utilize frequency analysis to determine the original letters by looking at how frequently they appear in the ciphertext.

4. CONCLUSION

Using the Google Colab program, this study was able to examine the shift password method's cryptographic security. According to the study's findings, the shift password has serious vulnerabilities when it comes to frequency analysis and brute force attacks. As a platform for cryptographic security implementation, simulation, and analysis, Google Colab has shown to be efficacious. Given its simplicity and versatility, Google Colab is advised as a platform for additional cryptography experimentation. This study highlights the need for a more secure way to protect sensitive data while also showing that the shift password is a weak and that Google Colab has potential as a tool for cryptographic research. Nonetheless, it is strongly advised to employ a more powerful and sophisticated cryptographic algorithm in light of these flaws.

REFERENCES

[1] N. H. Sabbry and A. B. Levina, "An Optimized Point Multiplication Strategy in Elliptic Curve Cryptography for Resource-Constrained Devices," *Mathematics*, vol. 12, no. 6, 2024, doi: 10.3390/math12060881.

[2] A. Verma, N. Kaur, and C. Jhanjeri Mohali, "A Comparative Study of Classical Substitution Ciphers," *Int. J. Eng. Res. Technol.*, vol. 3, no. 9, pp. 360–364, 2014, [Online]. Available: www.ijert.org

[3] A. Sugiarto, "Rancang Bangun Tracking Senjata SS2 Pada Drone Quadcopter S2GA," *J. Tek. Elektro dan Komput. TRIAC*, vol. 7, no. 1, pp. 1–5, 2020, doi: 10.21107/triac.v7i1.7276.

[4] F. Husaini, A. M. H. Pardede, and I. Gultom, "Penerapan Enkripsi Menggunakan Metode Elgamal guna Meningkatkan Keamanan Data Text dan Gambar," *JUKI J. Komput. dan Inform.*, vol. 4, no. 1, pp. 67–73, 2022, doi: https://doi.org/10.53842/juki.v4i1.104.

[5] K. A. Santoso, A. Pradjaningsih, and E. Delenia, "Pengaman Teks dengan Kombinasi Metode *Electronic Code Book* (ECB) dan Kode *Seven Segment Display*," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 11, no. 1, pp. 85–94, 2024, doi: 10.25126/jtiik.20241117448.

[6] D. I. Mulyana and A. Pratama, "Optimasi Deteksi Pengenalan Huruf Hijaiyah Dengan Metode Tepi Canny Dan Morfologi," *INTECOMS J. Inf. Technol. Comput. Sci.*, vol. 6, no. 2, pp. 717–725, 2023, doi: 10.31539/intecoms.v6i2.7663.

[7] R. Safitri, P. W. Prasetyo, D. E. Wijayanti, S. Arifin, F. Setyawan, and J. Repka, "Text security by using a combination of the vigenere cipher and the rubik's cube method of size 4×4×4," *Al-Jabar J. Pendidik. Mat.*, vol. 14, no. 2, p. 281, 2023, doi: 10.24042/ajpm.v14i2.14276.

[8] Yusfrizal, "Rancang Bangun Aplikasi Kriptografi Pada Teks Menggunakan Metode Reverse Chiper Dan Rsa Berbasis Android," *JTIK (Jurnal Tek. Inform. Kaputama)*, vol. 3, no. 2, pp. 29–37, 2019, [Online]. Available: http://jurnal.kaputama.ac.id/index.php/JTIK/article/view/173

[9] A. A. Permana and D. Nurnaningsih, "Application of Cryptography With Data Encryption Standard (Des) Algorithm in Picture," *JIKA (Jurnal Inform.*, vol. 4, no. 2, p. 9, 2020, doi: 10.31000/jika.v4i2.2640.

[10] S. Schmeelk and L. Tao, "A Case Study of Mobile Health Applications: The OWASP Risk of Insufficient Cryptography," *J. Comput. Sci. Res.*, vol. 4, no. 1, pp. 22–31, 2022, doi: 10.30564/jcsr.v4i1.4271.

[11] W. Vallejo, C. Díaz-Uribe, and C. Fajardo, "Google Colab and Virtual Simulations: Practical e-Learning Tools to Support the Teaching of Thermodynamics and to Introduce Coding to Students," *ACS Omega*, vol. 7, no. 8, pp. 7421–7429, 2022, doi: 10.1021/acsomega.2c00362.

[12] R. Nazar, "Implementasi Pemrograman Python Menggunakan Google Colab," *J. Inform.*

dan Komput. , vol. 15, no. 1, pp. 50–56, 2024.

[13] G. I. E. Soen, M. Marlina, and R. Renny, "Implementasi Cloud Computing dengan Google Colaboratory pada Aplikasi Pengolah Data Zoom Participants," *JITU J. Inform. Technol. Commun.*, vol. 6, no. 1, pp. 24–30, 2022, doi: 10.36596/jitu.v6i1.781.

[14] Uci Julya Ningsih, Sophia Salsabila, Isniar Hutapea, Dewi Santika, and Indra Gunawan, "Pendekripsian Caesar Chiper Dengan Menggunakan Teknik-Teknik Kriptanalisis," *J. Ilmu Komput. dan Multimed.*, vol. 1, no. 1, pp. 11–15, 2024, doi: 10.46510/ilkomedia.v1i1.10.

[15] P. Wirayudha, D. Widiatmoko, A. Sridaryono, M. Syafaat, and K. Kasiyanto, "Pemanfaatan Modul Lora SX1278 Sebagai Sistem Telekontrol pada Robot Penjaga," *Reslaj Relig. Educ. Soc. Laa Roiba J.*, vol. 6, no. 3, pp. 2201–2211, 2024, doi: 10.47467/reslaj.v6i3.6102.

[16] D. R. Saragi, J. M. Gultom, J. A. Tampubolon, and I. Gunawan, "Pengamanan Data File Teks (Word) Menggunakan Algoritma RC4," *J. Sist. Komput. dan Inform.*, vol. 1, no. 2, p. 114, 2020, doi: 10.30865/json.v1i2.1745.