



An Encrypted QR Code Using Layered Numeral Calculation for Low Powered Devices

Rafina Destiarti Ainul¹, Susilo Wibowo² and Irzal Zaini³

^{1,2,3} Electrical Engineering Department, Universitas Surabaya (UBAYA)

Raya Kalirungkut, Surabaya 60293, East Java, Indonesia

{rafina, susilo_w}@staff.ubaya.ac.id, irzal.zaini00@gmail.com

Abstract. Providing security system for every electronic data exchange through internet as the unsecured medium has become an essential regulation. Conventional Caesar Cipher had less computation complexity than other security method that really appropriate with low powered device requirement. However, it is susceptible attack by brute force attack cryptanalysis due to its simplicity calculation. Therefore, this paper proposed enhancement scheme for Caesar cipher using layered numeral calculation based on expansion, transposition, multiplication combination. Adding several processes can be provided difficult cipher and also eliminated the weakness of Caesar cipher. In this paper, cipher text output from enhanced Caesar encryption is encoded into QR code as the encrypted QR code which can be employed text data protection. According to the experiments used ESP32, this enhanced Caesar cipher only consumes about 2.34 ms which will not burden the devices while doing other processing and minimizing power usage.

Keywords: Enhanced Caesar Cipher, Encrypted QR Code, Processing Time.

1 Introduction

In recent time, collect data and identify object automatically without human involvement became a popular technology developed by many researchers. Quick Response (QR) code technology as one example of this technology is stored data information in the form of two dimensional matrix bar-codes. Matrix barcode of QR code using black and white pattern to encode bit of data information which was represent numerical data in the text [1-3]. Using QR code technology, various mobile applications can exchange data information easily, such as: authentication in unmanned library, gate access control of buildings, and payment system in retail store [1]. Several applications, using QR code for sharing personal confidential information [2]. Unfortunately, QR codes are vulnerable to security risk due to public placement and occasionally even illegally used. To address this issues, there are several previous work propose a novel QR code encryption system. In [1], add randomized rotation for securing QR code. While [2] is using multi-layer encryption system based on image's mathematical processing method. The two previous works used an image processing approach

© The Author(s) 2023

M. Hartono et al. (eds.), *Proceedings of the 4th International Conference on Informatics, Technology and Engineering 2023 (IncITE 2023)*, Atlantis Highlights in Engineering 21,

https://doi.org/10.2991/978-94-6463-288-0_50

that required a long processing time. This security approach is not suitable for low powered devices necessary.

According to various encryption techniques, Caesar cipher is one of the most ancient and simple cipher techniques [4]. As classical techniques, Caesar cipher is easily broken by Brute force attacker [4-16]. Caesar cipher is only use 26 keys value for encryption process. Attacker can use all the 26 possible set of keys for decrypting the data which is highly unsafe for data transmission particularly through internet [5]. By considering the weakness of Caesar cipher, many researchers have designed to improve security level of existing Caesar cipher. Several researchers using two approaches for rectifying Caesar cipher method, i.e. combination with another method and modification the scheme or key. Previous work in [9,13][14-15], combining Caesar cipher with another classic method such as Vigenere, Affine, Playfair cipher can provide high security and also maintaining the confidentiality of messages. This combination make the encoding process is more difficult to solve. While, some modification for Caesar Cipher have been done in [4-8][9-11]. Its modification is based on key transformation into binary data which allowed the numbers of key combination are more than 26. The binary key data arrangement can be used transposition or shifting based its time released, matrix formation, rectangular method or its desired pattern which aims to complicate the process of encryption and decryption at Caesar cipher.

However, the enhancements of Caesar cipher from previous works were not implemented at low powered device that couldn't be analyzed the processing time requirement. At the [4] have been proposed extended Caesar ciphers as called as ECC for low powered devices using more mathematical operation in key generation and encryption process. The key generation process is taken in binary form and added with its factorial function. Then, the encryption process is XOR of key and its plaintext. The result show that, adding these schemes to Caesar cipher has higher avalanche effect and more equalization in the frequency test [4]. Unfortunately, the character length of this technique will change according to the size of plaintext and key. This condition is not suitable with the requirement of QR code generation. Generating QR code in 21 x 21 dot data, require 17 bytes data or equivalent with 17 characters.

Therefore in this paper we propose another technique for enhancing the Caesar cipher algorithm. Using simple mathematical calculation which is obtained by substitution, transposition, expansion and multiplication, can produce low processing time and higher security level than conventional Caesar cipher technique. The size of cipher text can be adjusted with the requirement of QR code generation. The contribution of this paper is designed low power usage of Encrypted QR code based on developing the Caesar cipher algorithm with layered numeral calculation which is also implemented at ESP32 pi device. The proposed security is one solution for low powered device and still maintaining the confidentiality of data in QR code form. All data transmission of this system is through Wi-Fi of ESP32 device.

The rest of paper is organized as follows: the adopted algorithm is discussed in Section II, which is involved the conventional algorithm of Caesar cipher. The proposed mechanism of encrypted QR code with the enhanced Caesar cipher is presented

in Section III. The experimental result to verify the proposed algorithm is analyzed in Section IV. Finally, we draw the conclusion in Section V.

2 Adopted Conventional Caesar Cipher

In this section, we describe adopted algorithm using conventional Caesar cipher. Historically, Julius Caesar as the inventor of the Caesar cipher method offered additive cipher for securing his communication. Caesar cipher has 26 set of key to encrypt data according to the alphabet numbers. Each character of the alphabet is marked with a number in order in which it is assigned, as shown at Fig.1. These value numbers was used for the reference number of shifting position after the encryption process. Caesar cipher is sometimes referred as “shift chipper”.

Character	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Value	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Fig. 1 The converted value of Alphabet Character at Caesar cipher

The plaintext as the original message would be replaced by the other character based its shifting key generation. For instance, with a shifting key 5, B would be replaced by G, and R would be replaced by W and so on. These replacement characters are the cipher text result from Caesar cipher encryption. Then, to facilitate the shifting process of Caesar cipher, it can be written using additive formula as [4-6]:

$$C = E(k, P) = (P + k) \bmod 26 \tag{1}$$

$$P = D(k, C) = (C - k) \bmod 26 \tag{2}$$

C as the cipher text message are encrypted using E () function which is formed by additive process of Plaintext (P) with the key number (K) then calculate the modulus result. The modulus value is 26 due to its characters number as well as key probability of Caesar cipher. While, at the decryption process D (), the cipher text are minus with the key number. This process was calculated using converted value of the plaintext refers to Fig.1.

3 Construction of Proposed System

In this section, we describe our proposed system construction of encrypted QR code using enhanced Caesar cipher algorithm which are consist of network model deployment and proposed algorithm of enhanced Caesar cipher. The network model deployment will be explained the real implementation of this system using ESP32 as the low powered device equipped by Wi-Fi for local data transmission and internet for secret key exchange. While the subsection of enhanced Caesar will be explained the several numeral calculations for improving the conventional Caesar cipher algorithm.

3.1 Network Model

In this system, ESP32 as the low powered devices have two roles, i.e. the QR code scanner send the data to the ESP32 master via Wi-Fi and ESP32 master upload the data to the database via internet connection. While, the main items for generating key is initialized by network administrator. At the beginning of this system work, network administrator uploaded the key via internet to real time database. The limitation of this paper ignored the security scheme of internet connection due to assumed with TLS protocol.

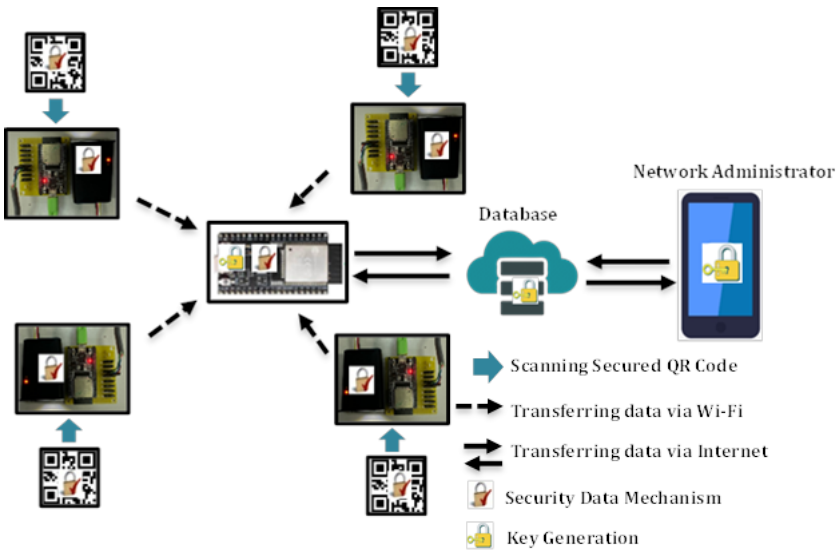


Fig. 2 Topology system of proposed encrypted QR Code

This paper is only focused at low power security scheme. ESP32 master distributed the key to the other ESP32 scanner which are equipped by proposed security scheme. After receiving the key, ESP32 scanner can create the encrypted QR code. In the implementation process, each encrypted QR code card at the size of 21 x 21 dot data, will be scanned and directly send to the master. ESP32 master can decrypt the encrypted QR code and stored the data at database. So hope this network can be applied for some fields such as security payment at department store, bridge game competition and the other application with encrypted QR code requirements.

As shown at Fig.2, there are four ESP32 scanners and one ESP32 master for analyzing the influence of security algorithm addition to its parallel processing. ESP32 as the low powered device has particular limitation when decryption processing and multi receive data are done simultaneously. Physically, the ESP32 device will be very hot and will be affected to its performance especially at wireless transmission which required stable transmit power. Therefore, in this paper will be analyzed the processing time of this process that is directly related to its power usage.

3.2 Enhanced Caesar Cipher

In this system, we propose enhanced Caesar cipher with some numeral calculation combinations. This proposed algorithm can be used for securing the QR code data and sharing the secret key based on the network model scenario. As seen at Fig.3, there are 4 main processes at this scheme consist of mapping data using Caesar, shifting the data using interleave, multiplying and expanding the data for getting required data size. In reverse, this process will be run on encryption and decryption phase. Using multi-layer numeral calculation for securing the original message can be improved the security level from Caesar cipher.

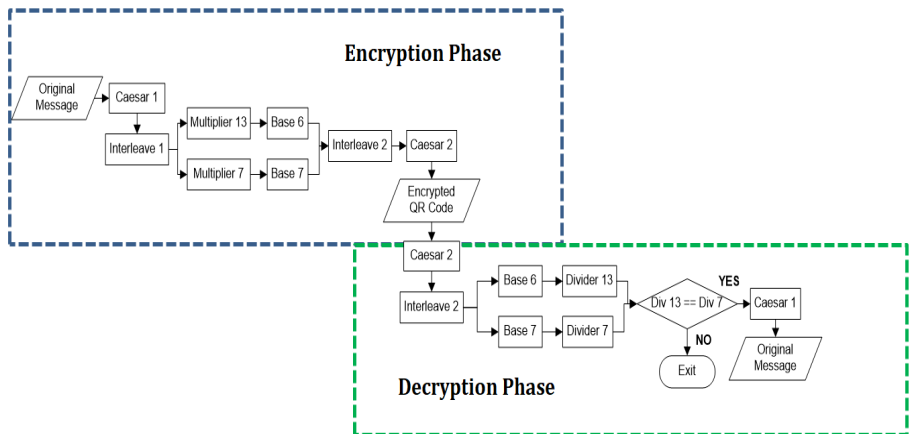


Fig. 3 Proposed security scheme of enhanced Caesar cipher

This proposed security schemes hope it can be implemented for bridge game application. Therefore, there are 152 lists of plain text code which are formed by playing card code, bidding code and board code. Writing the Original Code consists of 2 characters, for example: as Curly Card is written 'CA', which means Club Ace and so on. First step for creating the encrypted QR code is mapping the original code into six decimal number characters with 6 bytes size as shown to this following formula:

$$S2 (plain\ text) \rightarrow 1\ 2\ 3\ 4\ 5\ 6 (cipher\ text = C_1) \tag{3}$$

This mapping process can be used same number value with different permutation or used different number value with same or different permutation poles. The mapping process is adopted the Caesar cipher algorithm that were encrypted plaintext using shifting as well as permutation process in this step. Then, C_1 is reprocess again using interleave based on transposition pole. The transposition pole will be saved as key, derived as:

$$1\ 2\ 3\ 4\ 5\ 6 (C_1) \rightarrow 4\ 6\ 2\ 1\ 3\ 5 (C_2) \rightarrow key = (4\ 6\ 2\ 1\ 3\ 5) \tag{4}$$

The interleaving 1 result (C_2) will be multiplied with 13 and 7 value numbers. This value number is determined by experimental process for determining numbers that can be used for decrypting process and getting the original code. After that, the result from multiplier 13 is converted to numerical base 6, whereas the multiplier 7 outputs is converted into base 7, calculated as:

$$\begin{aligned}
 C_2 \rightarrow 462135 \times 1 &= 6007755 (C_3) \rightarrow \text{base6} \rightarrow C_5 = 332433403 \\
 C_2 \rightarrow 462135 \times 7 &= 3234945 (C_4) \rightarrow \text{base7} \rightarrow C_6 = 36332220
 \end{aligned} \quad (5)$$






C_5 and C_6 are combined at the second of interleaving steps with a result of 17 value numbers (C_7). Then, the last encryption process is transformed 17 value numbers into alphabet as well as Caesar Cipher method.

$$C_5|C_6 \rightarrow \text{Key}_{(5,9,10,6,11,12,7,8,1,13,14,2,15,16,3,17,4)} C_7 = 23322420333640333 \quad (6)$$

$$C_7 \rightarrow \text{Caesar}_2 = C_8 = CDDCCECADDDGEADDD \quad (7)$$

QR code at the size of 21 x 21 can relocate 17 maximum characters. Hence, in this proposed method is designing encryption process with 17 characters output. The final cipher text result can be decoded using QR code generator that formed as encrypted QR code. The results of encrypted QR code from five example of original code are listed at Table 1.

Table 1. Five examples of encrypted QR codes

No.	Original Code	Final Cipher text (C_8)	Encrypted QR Code
1	S2	CDDCCECADDDGEADDD	
2	S3	BDCBFFGADFBFFCGAC	
3	S4	GECAADGAE EEF AFDA A	
4	S5	EEBABA AAECAEDCBAF	
5	S6	BFAGBEA AFBCDEFFAE	

Decryption process is doing same process as the encryption process. The main concern at decryption process is the pair secret key that should be same with the encryption process. The reverse from multiplier at encryption will be changed as divider 13 and divider 7. The result from each divider should be same, if the result is different, the original code can't be determined.

4 Implementation and Experimental Measurements

In this section, we describe the real implementation of encrypted QR code using ESP32 device via Wi-Fi link communication. The length of data wouldn't be influenced to this system; due to static result of cipher text should be in 17 characters or equivalent to 17 bytes of data. The successful system will be proved to the performance result involve the examination of data security process and security scheme evaluation. There are some parameter specifications of ESP32 as the low powered devices which can be influenced to the performance result. The specification of low powered device and the software addition of this paper are listed at Table 2.

Table 2. Device specification

Devices	Specifications
ESP32-WROOM-32UE	<ul style="list-style-type: none"> ▪ Networking Bluetooth V4.2 BR/EDR and Bluetooth LE specification, Wi-Fi 802.11b/g/n. ▪ CPU ESP32-D0WD-V3 or ESP32-D0WDR2-V3 embedded, Xtensa dual-core 32-bit LX6 microprocessor, up to 240 MHz, Operating voltage/Power supply: 3.0 ~ 3.6 V.
Software	
QR Code generator → Barcode.tec.it ESP32 → Arduino IDE Database → Google Firebase Mobile Apps → Android Studio	

4.1 Examination of Data Security Process

The work flow of this system is started from key generation from network administrator via mobile apps. Admin generate pair of secret key for both of interleaving process. The secret key is encrypted using default TLS protocol via internet. ESP32 master should be downloaded first the pair of secret key at database. If the downloading key process is successful, then ESP32 master will be distributed the key into each of ESP32 scanner. In offline phase, user can be prepared the barcode card which has been adjusted with encryption result. The secured QR code is scanned to each ESP32 slave, and then it is decoded by ESP32 slave in the form of cipher text. Each ESP32 slave forward the cipher text to the ESP32 master. The cipher text is decrypted at

ESP32 master using the pair of secret keys that have been share before from the administrator. Administrator can be observed the original code from the database which is uploaded by ESP32 master.

In this system, we evaluate some kind of processing time parameter which is influenced to the security process performance. Preparation time is a process to initialize some required parameter such as key generation, port, and IP address, decode the encrypted QR code. Due to the different task of each ESP32 device, processing time is classified into encryption time and decryption time. The processing time is measured 10 times the experiment for each device. The processing time result at Table 3 is average result from 10 times measurement data collection.

Table 3. Processing time measurement of proposed encrypted QR code scheme

Encryption Process							
Devices Role	Prepare Time (ms)	Caesar1 (ms)	Interleave1 (ms)	Base & multiplier (ms)	Interleave2 (ms)	Caesar2 (ms)	Total (ms)
ESP32 Slave-1	2340	0.11	0.21	0.4	0.36	0.12	2341.2
ESP32 Slave-1	2332	0.09	0.31	0.32	0.34	0.14	2333.2
ESP32 Slave-1	2582	0.10	0.27	0.35	0.32	0.15	2583.19
ESP32 Slave-1	2346	0.11	0.28	0.37	0.31	0.13	2347.2
Average	2400	0.1025	0.26	0.36	0.33	0.135	2401.198
Decryption Process							
Devices Role	Prepare Time (ms)	Caesar2 (ms)	Interleave2 (ms)	Base & divider (ms)	Interleave1 (ms)	Caesar1 (ms)	Total (ms)
ESP32 Master	5970	0.13	0.21	0.47	0.39	1.11	5972.31

According the processing time result of decryption process is takes longer than encryption process due to the key generation as the preparation time measurement is downloaded first from database. The downloading process is dependent to the internet connection. Whereas the preparation time at ESP32 slave is only used for Wi-Fi connection set up without key generation which was distributed before from ESP32 master. While at the other measurement scenarios, preparation time at ESP32 slave without any encryption process is only for setting up the Wi-Fi connection and decodes the encrypted QR code. It takes a time up to 2105ms that is almost same with first scenarios at 2400ms. As well as processing time from each step of this proposes scheme is requires only less than 1ms except the Caesar1 at decryption process. This is due to the addition of comparing cipher text result for authenticating and validating the original code. The authentication at Caesar1 checks and compares the result from

the divider phase, if the result of divider 7 same with divider 13, the original code can be decrypted. The results prove that, the proposed encrypted QR code requires lower processing time less than 7ms. The processing time is relatively fast will have small chance in consuming a lot of power resource. Therefore this scheme is appropriate for low powered device when in the future it will be developed with the other processing system.

4.2 Security Schemes Evaluation

In this paper, we evaluate the security strength using two scenarios, first using attacker device as the fake ESP32 slave which can be connected to this network system. The ESP32 slave pretends as the scanner to scan the fake secure code. When the fake secure code is received by valid ESP32 master, the valid ESP32 master can't be decrypted because the secret key pair from interleaving process is different. But if the valid node can decrypt the code, the length of cipher text message is more than 17 characters. At network administrator will be monitored continuously, when the original code is entered more than desired interval, at that time key generation will be updated. The second evaluation is using fake encrypted QR code. When the fake QR code is scanned at the slave scanner, the scanner will be forwarded to the ESP32 master, but decryption process is failed due to the encoding result have different format data. The common people will see that the shape of QR code is always the same, whereas each encrypted QR code has different pattern and different code.

Our implementation of enhanced Caesar cipher using layered numeral calculation for low powered devices satisfies privacy and data integrity using ESP32 hardware. There are many implementation systems for improving the Caesar cipher algorithm. According to the previous works [4-13] didn't evaluate the real processing time at some hardware, whereas processing time is the major parameter for evaluating power usage consumption. Meanwhile, by different approach for giving the innovative enhancement of Caesar cipher [14], measured its proposed scheme required up to 2.098 seconds execution time in 20 bytes size of data. Its measurement process was not used low powered device, which was takes longer execution time than our proposed method. It is different with research in [16] that used combination from blowfish, Caesar cipher and DES with java algorithm for protecting various types' media files such as jpg, png, mp4, etc. Using java algorithm in this research is impossible do to its large memory and processing time requirement. Using steganography method for protecting QR code data also proposed in [3] which are not also appropriate for this system. This condition is not compatible for our system that is required fast processing time for minimizing power consumption.

5 Conclusion

In this paper, we propose encrypted QR code for low powered devices. This security scheme is provided enhancement of Caesar cipher which is equipped with privacy data integrity and layered security with smaller processing time. Combining layered numeral calculation using transposition, expansion, multiplier and divider can im-

prove complexity level at Caesar cipher. Renewal key generation scenario from network administrator can also minimize cryptanalysis and brute force attack in Caesar cipher. The performance of processing time result show that using ESP32 as the low powered devices is achieved less than 1 milliseconds for encrypting or decrypting the code, while all system processing time including decoding QR code, key generation and sending code is only required up to 8.37 seconds. In our future work, enhancement using modern cryptography algorithm for low powered devices will be proposed.

References

1. S. S. R. Garnaik, Y. Kim and J. Ryoo, "SQR: Encrypted QR Transaction with Randomized Rotation," 2022 13th International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea, Republic of, 2022, pp. 1697-1702, doi: 10.1109/ICTC55196.2022.9952603.
2. A. F. M. Fauzi, N. N. Mohamed, H. Hashim and M. A. Saleh, "Development of Web-Based Smart Security Door Using QR Code System," 2020 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS), Shah Alam, Malaysia, 2020, pp. 13-17, doi: 10.1109/I2CACIS49202.2020.9140200.
3. Haroon Rashid Hammood Al Dallal and Wijdan Noaman Marzoog Al Mukhtar, "A QR Code Used for Personal Information Based On Multi-Layer Encryption System", *Int. J. Interact. Mob. Technol.*, vol. 17, no. 09, pp. pp. 44–56, May 2023.
4. Priya Verma and Gurjot Singh Gaba, "Extended Caesar Cipher for Low Powered Devices", *Int. J. of Control Theory and Applications*, vol. 9. No.11, pp. 5391-5400, 2016.
5. Priya Verma and Gurjot Singh Gaba and Himanshu Monga, "Modified Cesar Cipher using Rectangular Method for Enhanced Security", *Journal of Comm. Technology, Electronics and Computer Science*, issue 8, pp.1-5, 2016.
6. Benni Purnama, A.H. Hetty Rohayani, "A New Modified Caesar Cipher Cryptography Method with LegibleCiphertext From a Message to Be Encrypted", *Procedia Computer Science*, Vol. 59, pp. 195-204, 2015.
7. R. Majumder, S. Datta and M. Roy, "An Enhanced Cryptosystem Based on Modified Classical Ciphers," 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2022, pp. 692-696, doi: 10.1109/ICACCS54159.2022.9785033.
8. S. Y. . Wulandari, "Cryptography: A Combination of Caesar and Affine Cipher to Conceal the Message", *icse*, vol. 3, pp. 741–744, Apr. 2020.
9. W. Haryono, "Comparison Encryption of How to Work Caesar Cipher, Hill Cipher, Blowfish and Twofish", *Data Science: J. of Computing and Appl. Informatics*, vol. 4, no. 2, pp. 100-110, Jul. 2020.
10. Alisawi, W.C.; Oleiwi, Z.C.; Alawsy, W.A.; Alfoudi, A.S.; Hadi, N.K. Improvement of Classical Cipher Algorithm based on a New Model of Timed-Released Encryption. *Int. J. Appl. Eng. Res.* 2019, 14, 3531–3536.
11. Rahim, R, et. al., "Enhancement three-pass protocol security with combination caesar cipher and vigenere cipher", *Journal of Physics: Conference Series*, Series 1402, 2019.
12. M. D. Sinaga, N. S. B. Sembiring, F. Tambunan and C. J. M. Sianturi, "Hybrid Cryptography WAKE (Word Auto Key Encryption) and Binary Caesar Cipher Method For Data

- Security," 2018 6th International Conference on Cyber and IT Service Management (CITSM), Parapat, Indonesia, 2018, pp. 1-5, doi: 10.1109/CITSM.2018.8674346.
13. D. Gautam, C. Agrawal, P. Sharma, M. Mehta and P. Saini, "An Enhanced Cipher Technique Using Vigenere and Modified Caesar Cipher," 2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2018, pp. 1-9, doi: 10.1109/ICOEI.2018.8553910.
 14. S. N. Gowda, "Innovative enhancement of the Caesar cipher algorithm for cryptography," 2016 2nd International Conference on Advances in Computing, Communication, & Automation (ICACCA) (Fall), Bareilly, India, 2016, pp. 1-4, doi: 10.1109/ICACCAF.2016.7749010.
 15. S. Karthiga* and Dr. T. Velmurugan, "Enhancing Security in Cloud Computing using Playfair and Caesar Cipher in Substitution Techniques," International Journal of Innovative Technology and Exploring Engineering, vol. 9, no. 4. Blue Eyes Intelligence Engineering and Sciences Engineering and Sciences Publication - BEIESP, pp. 912–920, Feb. 28, 2020. doi: 10.35940/ijitee.d1363.029420.
 16. S. Saudagar, N. Kamtalwar, H. Karadbhajne, M. Karmarkar, H. Kendre and O. Ketkar, "File Encryption-Decryption using Java," 2023 International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Bengaluru, India, 2023, pp. 855-859, doi: 10.1109/IDCIoT56793.2023.10053514.

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

