University of Baghdad
College of Engineering
JOURNAL OF ENGINEERING

Journal of Engineering

journal homepage: www.jcoeng.edu.iq

Volume 30 Number 11 November 2024

Evaluation of Rijndael Algorithm for Audio Encryption by Brute Force Attack

Sajaa G. Mohammed*, Nuhad Salim Al-Mothafar

Department of Mathematics, College of Science, University of Baghdad, Baghdad, Iraq

ABSTRACT

The use of information transfer is a major reason for the spread of information piracy, especially digital information that includes audio or other information, to preserve data. Encrypting voice messages is considered one of the most secure ways to protect information due to the difficulty of disclosing it. Our algorithm is robust, effective, efficient, and has security capabilities due to the different key lengths and block sizes (128 bits, 192 bits, or 256 bits). The Rijndael algorithm is an effective tool for encoding audio files and has won many awards. The Rijndael algorithm is compatible with many different audio formats. We will use it to encrypt WAVE files. The first step is to generate keys using the proposed method consisting of key lengths and block sizes (128 bits, 192 bits, or 256 bits). We convert the audio data into a 256 sample. After that, the cipher text will be generated for us. Once we encrypt the audio file, we will have several audio files. We will then use a proactive approach to evaluate the success of this initiative through a brute force attack to break the code. The Rijndael algorithm plays a role in this study by demonstrating the effectiveness of encryption, Reijndeal encryption, is designed to withstand brute force attacks, employs a strong key, proper encryption algorithm, secure hardware, and software to prevent side-channel attacks, and thwart cryptanalysis attempts. It also offers a cutting-edge perspective. Its primary goal is to protect data across the digital landscape during transmission, storage, and protection processes.

Keywords: Rijndael, Audio ciphering, Block cipher Rijndael, Brute force attack.

1. INTRODUCTION

These days, audio data security plays a major role in the IT business and is expanding quickly **(Kakde et al., 2015; Alabaichi et al., 2022; Mahmood et al., 2024).** Encrypting audio data prevents illegal access and manipulation while guaranteeing its integrity and confidentiality **(El Hanouti and El Fadili, 2021; Esttaifan, 2023; Abdul-Jabbar et al., 2023; Al-Yasiri,**

*Corresponding author
Peer review under the responsibility of University of Baghdad.
https://doi.org/10.31026/j.eng.2024.11.08

Article received: 11/03/2024
Article revised: 15/05/2024
Article accepted: 27/05/2024
Article published: 01/11/2024

2024; Yaro, 2024). Technology preservation of audio files from hackers and eavesdroppers became crucial for the tech professional. Thus, the necessity for swifter and more secure audio file encryption algorithms remains continual. Encrypting audio using cryptography involves simultaneously adding noise, or the key, to a plain text file. Decryption is using the same key to reveal the original plain text. Speech is an attractive hands-free human-computer interface broker that only requires basic hardware to purchase high-quality microphones and comes at a very low bit rate. Human speech is essentially recognized as continuous, connected speech without tedious practice (free speaker) since a vocabulary with the appropriate complexity (100,000 words) is incredibly difficult **(Sethia et al., 2016; Hassan et al., 2020).** However, algorithms, procedures, and techniques simplify the processing of voice signals and the recognition of spoken text by a speaker. To conduct audio file encryption, this algorithm uses Rijndael algorithm keys generated from speech audio files (WAVE). We tested and used the suggested algorithm **(Harba, 2018; Mohammed et al., 2021).** The Rijndael algorithm is a symmetric block cipher with lengths of 128, 192, and 256 bits that can handle data blocks of 128 bits. Of the restriction that the input and output sequences have the same length, the Rijndael encryption key, the input, and the output are all bit sequences of 128, 192, or 256 bits apiece. **(Aslam and Alkhaldi, 2015; Mohammed and Majeed, 2017).** Advanced encryption standard-based stenographic algorithm (AES). We first transform the sound into a picture before encrypting it using AES. Security assessments and simulations show how well the suggested algorithm performs **(Mohammed et al., 2021; Cheroiu et al., 2022)**. The research on Investigating Rijndael-based algorithms for Audio Ciphering approaches that were published from 2018 to 2020 is surveyed in this study. The subsequent sections further illustrate the present research orientations of the region. The following is a summary of this paper's contributions: In this article, we give a summary of the investigated Rijndael-based algorithms for Audio Ciphering.

In 2018, Audio Steganography was introduced by **(Mustafa et al., 2018).** This method was the Enhanced LSB Algorithm for High Secure. To ensure audio transmission was secure, an audio file was first encrypted using the Huffman method, then concealed using the AES technique, and then revealed using the cutting-edge LSB-Block algorithm. In 2019, the Audio Encryption Algorithm was introduced **(Kordov, 2019)**. This method was Permutation-Substitution Architecture. It used a pseudo-random number generator composed of a chaotic circle map and modified rotation equations into an audio form for the necessary Cryptographic security for audio file encryption. In 2020, Audio Encryption was introduced by **(Wang, 2020).** This method involved DNA coding and a chaotic system. It employed chaotic systems and DNA coding to confuse and disperse audio data into an audio file for high security. In 2020, Encrypt Audio File was introduced by **(Hassan et al., 2020).** This method uses a speech audio file as a key. It stores a two-secret key by converting a voice audio file to text. This text uses a hash function to generate a seed with two keys. The original audio file is then encrypted into the text using the Rijndael algorithm for strong audio transmission security.

This study aims to increase the complexity by using multiple block sizes and different key sizes (128 bits, 192 bits, or 256 bits). Although Rijndael audio encryption is generally considered secure, increasing the length to 256 bits means attackers cannot exploit it. We evaluate the work of this algorithm using a brute-force attack.

2. ENCRYPTION METHODOLOGY AND EVALUATION

This system design identifies four areas of work focus: the encryption and decryption process, the key generation method, the block size, and the method of evaluating the algorithm through a brute force attack **(Nițu et al., 2023)**. Information encryption hides information by using redundant cover data, including documents, audio files, movies, and images. Recently, the importance of this method has increased in many application areas. For example, encoding information in digital video, audio, and video involves using the human auditory system's limited capacity to encode the information **(Shakya and Lamichhane, 2016)**. To protect data privacy, the algorithm conceals all data entered within the audio. We created the system using the Brand-Rijdnal algorithm. This proposed system provides users with the ability to encrypt and decrypt data, as well as encrypt hidden information.

1. Introducing the method of using encryption
2. V.B. Studio used the Rijandnel algorithm, an algorithm that improves encryption accuracy and quality. The algorithm encodes them, embeds them in an audio file, transfers them to the destination along with text and other file formats, and evaluates their performance.

2.1 Audio Ciphering

Audio encryption is a way to protect data in a sound file from intrusive attacks that have applied the (noise) key and the exact algorithm to the original text. Our primary focus is on encryption techniques for handling audio data. We are analyzing and comparing fundamental encryption standards, such as AES, DES, and Triple DES (3-Data Encryption Standard), which demonstrate the ability to encode audio **(Ali and Rahma, 2015; Albahrani et al., 2021)**. The sound is WAVE format files frequently store digital audio data in Windows operating systems. In 1991, the Resource Interchange File Format (RIFF) component was first used for photos and videos. The descriptor chunk, the format chunk, and the data chunk are the three different sorts of chunks that make up the standard audio data format, WAVE. The format chunk contains important characteristics such as sample rate, byte rate, and bits per sample, whereas the descriptor chunk is the WAVE header. The data chunk includes raw data and specifies the size of the sound data. Experts generally recommend avoiding unfamiliar chunks as they may introduce new ones in the future **(Hassan et al., 2020; Masure and Strullu, 2023)**. This research focuses on aspects of previous chaotic audio algorithms that did not consider their advantages and disadvantages. Analogy and digital signal processing are important foundations for encryption. (**Ramalingam et al., 2020)** Have demonstrated the requirements for these algorithms, along with the security and statistical tests for their evaluation.

2.2 Rijndael Algorithm

NIST published a report summarizing all contributions and justifying the decision on October 2, 2000, when it officially announced that unaltered Rijndael would become the AES. NIST uses the following lines: This report concludes with a decision regarding the Rijndael. It appears that the Rijndael has always performed very well in both hardware and software across a wide range of computing data, regardless of its use in comments or not giving feedback. The key setting methods are excellent, and the key agility is good **(Joan et al., 2002)**. Daemen and V. invented the Rijndael algorithm, which was renamed the AES. **Fig. 1** depicts the steps involved in AES encoding and decoding. The Federal Information Processing Standard, or FIPS, 197, describes these actions, which include Sub Bytes, Shift Rows, Mix Columns, Add Round Key, Inv

Shift Rows, Inv Sub Bytes, and Inv Mix Columns. The final round differs slightly and does not include the Mix Columns action during encoding. To create a key schedule, the AES algorithm implements a key-extending routine. Given that an employee provides a key of 16, 24, or 32 bytes, it returns what is known as an expanded key of 16 × 11, 16 × 13, and 16 × 15 bytes, respectively **(Rathod and Gonsai, 2021; Li et al., 2023)**. Lastly, Rijndael's internal round structure appears to have good potential to benefit from instruction-level parallelism. Security is the most crucial category; even the most complex evaluations can only reveal a limited number of explanatory laws, The majority of them fall into the category that does not show any weakness, Rijndael operations are among the easiest operations to defend against hackers and timing attacks. In addition to this, it provides some definitions against such attacks without affecting performance in a very significant way. Rijndael encryption, designed to withstand brute force attacks, employs a strong key, a proper encryption algorithm, secure hardware, and software to prevent side-channel attacks and thwart cryptanalysis attempts **(Joan et al., 2002; Easttom, 2021).**

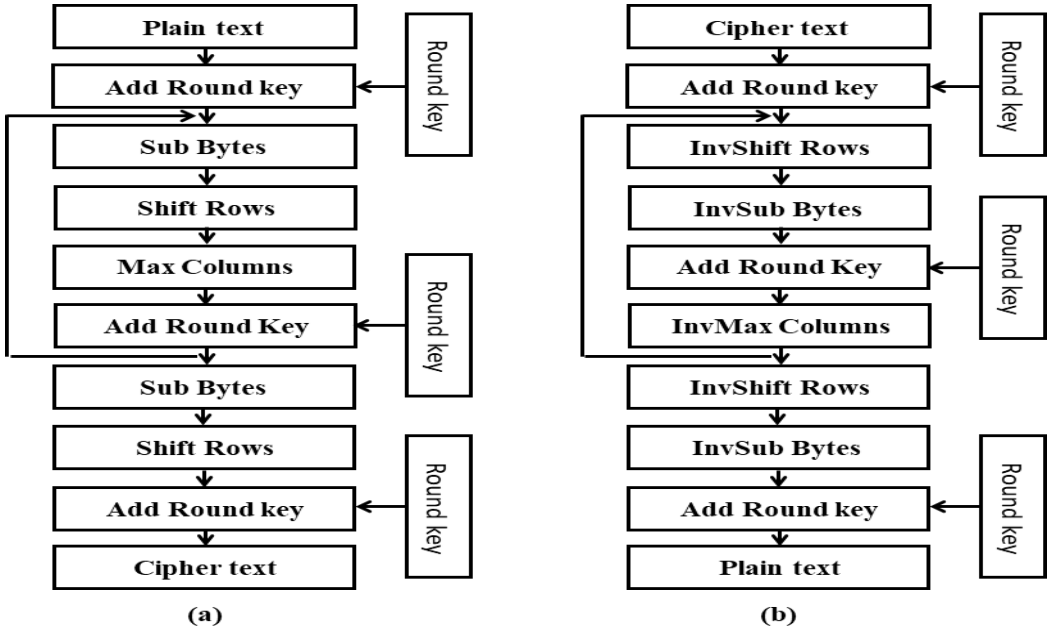

Figure 1. Rijndael algorithm (a) encryption structure; (b) decryption structure **(Li et al., 2023)**.

2.3 Block Cipher Rijndael Algorithm

The AES has three key standards: 128 bits, 192 bits, or 256 bits. Belgian cryptographers Daemen and Rijmen created the Rijndael cipher, which supports different block and key sizes, including 128, 160, 192, 224, and 256 bits. The AES standard, on the other hand, specifies a block standard of 128 bits and a key standard of 128, 192, and 256 bits. The Rijndael algorithm uses a substitution permutation matrix instead of a Feistel network The algorithm functions by placing the 128-bit plain text block into a 4-byte by 4-byte matrix, which changes as it progresses through its sequence of phases. We must convert this matrix, known as the state, to binary before inserting it into a matrix. The Rijndael cipher is a significant advancement in the field of ciphers, allowing for more flexible and secure encryption methods, as illustrated in **Table 1 (Daemen and Rijmen, 1998).**

Table 1. Convert the plain text block into binary.

11011001	01110010	10110000	11101010
01011111	00011001	11011001	10011001
10011001	11011101	00011001	11111101
11011001	10001001	11011001	10001001

Rijndael is an iterated block cipher with a variable block length and variable key length. It is possible to independently set the block and key lengths to 128, 192, or 256 bits. Rijndael, like Square and BKSQ, employs the wide trail technique in its design. This design technique offers protection from both differential and linear cryptanalysis. The approach breaks down the round transformation into various components, each with a distinct function **(Daemen and Rijmen, 1998; Raducanu, 2023).**

2.4 Rijndael Cryptographic Algorithm Method

The National Institute of Standards and Technology (NIST) in the United States developed the Advanced Encryption Standard (AES), also known as Rijndael, as the encryption specification used for electronic data. The substitution-arrangement network plan principle forms its foundation. The key standard can range from 128, 192, or 256 bits, with a block size of 128 bits. In the decryption algorithm, each phase of a round is the inverse of its corresponding stage in the encryption algorithm. These four basic processes apply to both encryption and decryption. The following are the four phases:

- Sub Bytes Transformation: Each byte is substituted with a different one in this non-linear substitution step based on the entries in a lookup table called an S-box. A one-to-one mapping of all byte values from 0 to 255, As shown in **Fig. 2.**, where r'(α, β) is the new value, and r (α, β) Original value as Eq. (1).

$$r'^{(\alpha,\beta)} = r\big(r(\alpha,\beta)\big) \quad (1)$$

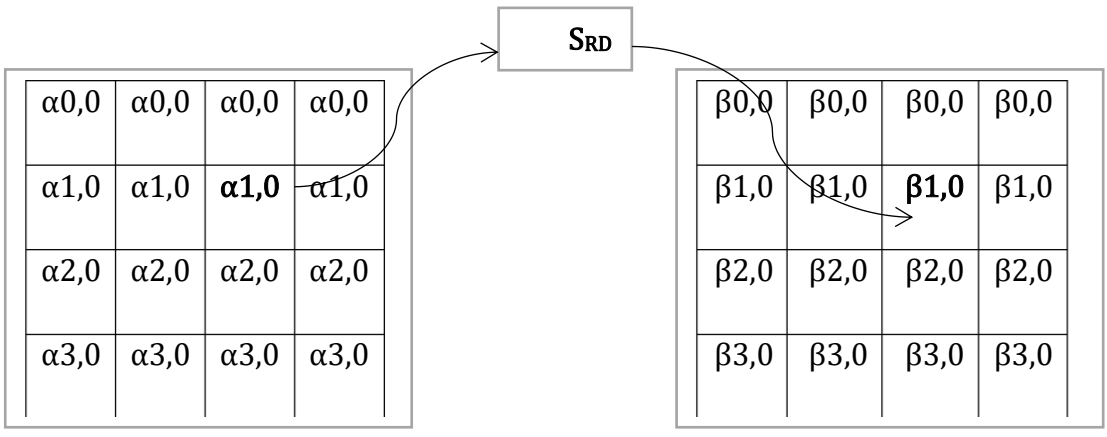

Figure 2. Subbytes operate on individual bytes of state**.**

- Shift Rows: This transposition phase involves cyclically shifting each state row a predetermined number of steps. As shown in **Fig. 3.** , where β is the row number, the rows are moved left by α certain number of bytes. where are(α, β) a new value, r (α, β) Original value as Eq. (2).

$$r'^{(\alpha,\beta)} = r(\alpha, (\alpha + \beta) mod\ 4) \quad (2)$$

Initial state

1α	1β	1γ	1δ
2α	2β	2γ	2δ
3α	3β	3γ	3δ
4α	4β	4γ	4δ

after shifted rows

1α	1β	1γ	1δ
2β	2γ	2α	2α
3γ	3δ	3α	3β
4δ	4α	4β	4γ

Figure 3. Shift rows.

- Mix Columns: The operation of a Mix Column is utilized following the application of the S-box and shift rows operation to the state. In this stage, the four bytes in each of the state's columns are combined by a mixing operation. A fixed polynomial, As shown in **Fig. 4. ,** μ is multiplied by the outcome as Eq. (3).

$$(x) = 3x^2 + x^2 + x + 2\ modulo\ x^4 + 1 \quad (3)$$

$$\begin{pmatrix} \beta0 \\ \beta1 \\ \beta2 \end{pmatrix} = \begin{pmatrix} 02\ 03\ 01\ 01 \\ 01\ 02\ 03\ 01 \\ 01\ 01\ 02\ 03 \\ 03\ 01\ 01\ 02 \end{pmatrix} \otimes \begin{pmatrix} \alpha0 \\ \alpha1 \\ \alpha2 \\ \alpha3 \end{pmatrix}$$

Figure 4. Shows the effect of the column mix step on the state.

- Add Round Key: A 4-word round key is provided for the first Add Round Key stage and each of the cipher's ten rounds using the Rijndael key expansion algorithm, which accepts a 4-word (16-byte) key as input. The first step is to copy the key into a group of four words. Next, four groups are created based on the values of the preceding four words. At last, the encryption text is obtained. As shown in **Fig. 4.** where r'(α, β) new value to an added key, r (α, β) Original value. k(i, j)Value of the key as Eq. (4).

$$r'(i,j) = r(\alpha,\beta) XOR\ k(i,j) \quad (4)$$

α0,0	α0,0	α0,0	α0,0
α1,0	α1,0	α1,0	α1,0
α2,0	α2,0	α2,0	α2,0
α3,0	α3,0	α3,0	α3,0

=

k0,0	k0,0	k0,0	k0,0
k1,0	k1,0	k1,0	k1,0
k2,0	k2,0	k2,0	k2,0
k3,0	k3,0	k3,0	k3,0

+

β0,0	β0,0	β0,0	β0,0
β1,0	β1,0	β1,0	β1,0
β2,0	β2,0	β2,0	β2,0
β3,0	β3,0	β3,0	β3,0

Figure 5. In the add round key, the round key is added to the state with a bitwise XOR.

All of the layers must be inverted to decode; that is, the Mix Column layer must become the Inv Mix Column layer, the Byte Substitution layer must become the Inv Byte Substitution layer, and the Shift Rows layer must become the Inv Shift Rows layer. On the other hand, there are certain similarities between the inverse layer operations and the encryption layer processes **(Shakya and Lamichhane, 2016; Easttom, 2021; Cheroiu, 2022).**

2.5 Brute Force Attack

However, Rijndael employs three distinct key lengths: 128 bits, 192 bits, and 256 bits. In the final scenario, a brute force attack method can attempt 2^{256} times to identify the utilized key, ultimately breaking the cryptosystem **(ElShafee, 2013)**. A straightforward approach, brute force problem solving often begins with the problem statement and definitions of the important concepts. The strategy's definition suggests that machine power, not human intelligence, is involved. The prescription for the brute force approach may also be stated as Just do it. In many cases, the brute force approach is also the simplest to implement. Take the exponentiation issue, for instance: compute α^{n} for a nonzero number a and a nonnegative integer n. Despite its apparent simplicity, this problem serves as a helpful example of many algorithm design techniques, such as the brute force approach. (Also note that computing α^{n} mod m for some large integers is a principal component of a leading encryption algorithm.) By the definition of exponentiation as Eq. (5).

$$\alpha^{n} = \alpha * \cdots\cdots * \alpha \tag{5}$$

This suggests simply computing αn by multiplying 1 by a n times. (Is it possible to name a few algorithms that you are already aware are based on the brute force method?). Despite its rarity as α source of intelligent or efficient algorithms, the brute force approach remains a crucial strategy for algorithm building. Firstly, brute force is applicable to a fairly broad range of issues, unlike certain other solutions. It appears to be the only general technique, in fact, for which identifying issues that it cannot address is more challenging. Secondly, the brute force method produces workable algorithms with no restrictions on instance size for a number of significant tasks, including sorting, searching, matrix multiplication, and string matching. Third, it might not be worth the investment to build a more efficient algorithm if a brute force approach can solve a problem at an acceptable speed and only a few instances of the problem need to be solved, then it might not be worth the investment to build a more efficient algorithm. Fourth, a brute force method can still be helpful for resolving small-scale examples of a problem, even if it is generally excessively inefficient. Ultimately, a brute force method can be a useful tool in theory or teaching by acting as a benchmark for more effective approaches to problem solving **(Majeed and Mohammed, 2017; Stinson, 2021).** A Brute Force approach to a problem involves looking through combinatorial items, such as combinations, subsets of a set, or permutations, for an element having a specific feature **(Kedem and Ishihara, 2009)**.

3. PROCEDURE STEPS FOR AUDIO CIPHER-BASED RIJNDEL ALGORITHM AND EVALUATION

The proof procedure for an evaluation audio cipher-based Rijndael algorithm in the Brute Force attack is shown in **Fig. 6.**

Algorithm (1): The proposed algorithm steps for audio cipher using the Rijndael algorithm.

Process: Begin

Step 1: Read the sound file (WAVE).
Step 2: Convert the sound file into hex data.
Step 3: Generate a key of the required length (128-bit, 192-bit, or 256-bit) encryption key.
Step 4: Use the Rijndael algorithm (AES) of the required length (128-bit, 192-bit, or 256-bit) block cipher to encrypt.
Step 5:To extract the decrypted file (WAVE) data, the receiver needs the new audio file (WAVE) with a different key and block cipher.
Step 6: Cryptanalysis of the Password cryptanalysis by Brute Force attack
Step 7: Return (S). End.

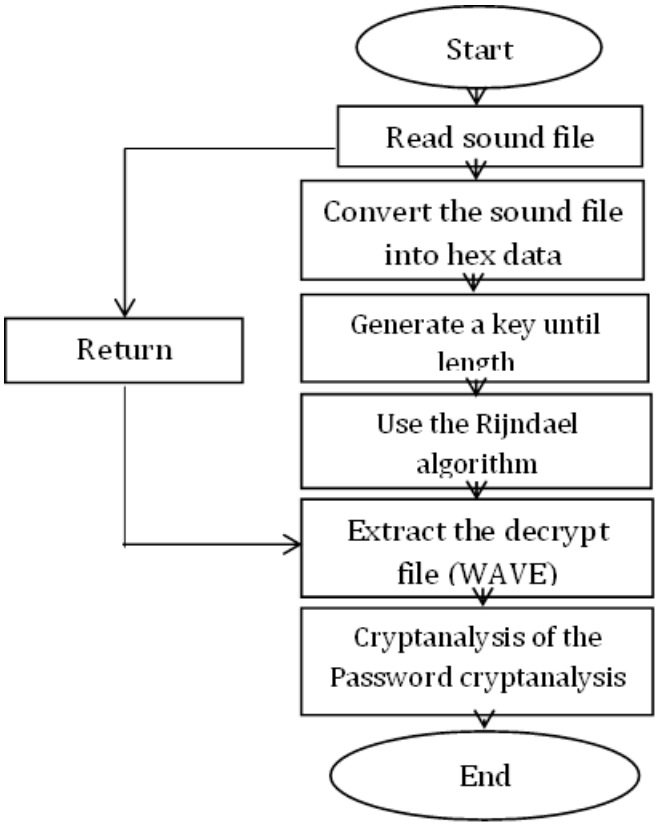

Figure 6. Audio encryption system with crack brute force attack.

4. RESULTS AND DISCUSSION

We implemented the proposed algorithm using the Visual Basic 6 driver We convert regular audio files with multiple block sizes and different key sizes (128-bit, 192-bit, or 256-bit) into encrypted audio files. Table 2 displays the results of the encryption and decryption operations using the Rijndael algorithms. In the first scenario, we successfully encrypted the audio while recovering the original data in plain text using a key size of 128 and a block size of 128. Next, we used a key size of 192 and a block size of 192 to arrive at a key size of 256 .256 and block size we used different encryption keys and block sizes in each situation to complete the decryption process. At the next stage, a brute-force algorithm attacks the algorithm to evaluate its performance, based on the number of permutations for each bit. The larger the size of the block and the key, the more difficult it will be. Therefore, at this stage, when the length increases, it will be difficult to break it. The ultimate goal of the study and the criteria by

which encryption algorithms are evaluated in terms of effectiveness and performance are the foundations on which we build our analysis of these results. See **Table 3** and **Fig. 7**. A snapshot of password cracking using the Brute force method which shows the number of permutations for each of 128, 192, and 256. Rijndael explains that the increased length of 256 bits, which allows access to the block and key size, is the cause.

Table 2. Results of the encryption and decryption.

Block Size	Key size	Plain	Encrypt	Decrypt
128	128	000102030405060708090A0B0C0D0E0F	E43763EEC1A8502433D6D520B1F0525	000102030405060708090A0B0C0D0E0F
128	192	000102030405060708090A0B0C0D0E0F	8046725C5FE415DC926CB08F54B1681A	000102030405060708090A0B0C0D0E0F
128	256	000102030405060708090A0B0C0D0E0F	0001020304050607080 90A0B0C0D0E0F	000102030405060708090A0B0C0D0E0F
192	128	000102030405060708090A0B0C0D0E0F1011121314151617	9F6DCA7965C74923C77A4A0A32FB43994C2E1B5BA0FB8035	000102030405060708090A0B0C0D0E0F1011121314151617
192	192	000102030405060708090A0B0C0D0E0F1011121314151617	1B83638F6919CA2C5D8B4999D13793E19180DAFDBDCA7372	000102030405060708090A0B0C0D0E0F1011121314151617
192	256	000102030405060708090A0B0C0D0E0F1011121314151617	88C27972B10BCCA2B4312B05CA87A541DBD6034A3AC070EC	000102030405060708090A0B0C0D0E0F1011121314151617
256	128	000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F	4D82D84A04FAA83D78D666369A8FC1FD611859EE3A7FF6C3B0D5C8D15E5737F5	000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F
256	192	000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F	EA79CCB5328288A7A3B24E537CC77E34A6AB17F0012C2CC04F303900BFC195EE	000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F
256	256	000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F	DC6214B7820FC68E844B42025A33C020CA578F2B73C80A48B220E6C0EE76EBBB	000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F

Table 3. Show the total perms and max perms.

Key size	Block Size	No. char	Total perms	Max perms
128	128	32char	6.51500091390582E+49	2821109907455
192	192	48 char	8.62489947332953 E+67	80318101764555
256	256	64 char	4.12662025496328 E+99	2176782333658496

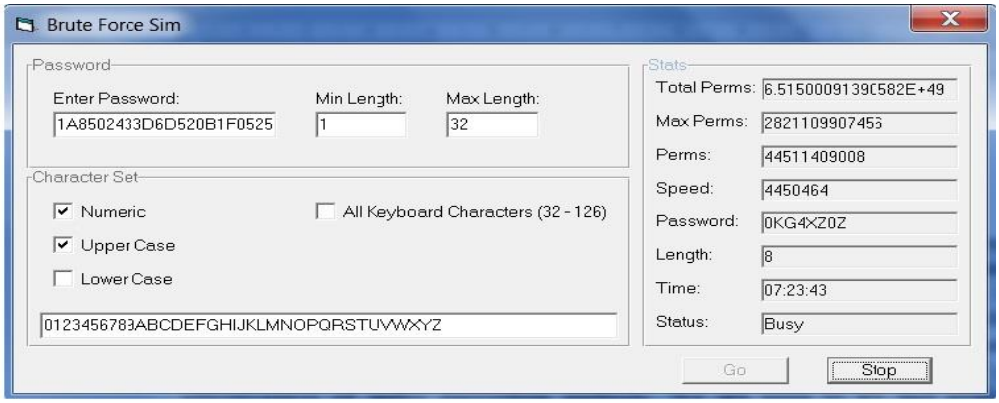

Figure 7. Snapshot of password cracking using the brute force method.

We previously knew that increasing length would affect decoding. This evaluation's success can be considered a very positive points. By comparing the key and block sizes used and their effect on sound quality, encoding strength, and restoration of the original audio, performance, and efficiency can also be examined. This figure shows us that 128 bits are used for both the key and the block size. We notice that the length of the encrypted key is 32 bits. We notice the number of permutations concerning the length of the 32 bits. The longer the encrypted key, the greater the number of permutations.

5. CONCLUSIONS

Cryptography, an ancient field, has evolved into a critical field of study for communications security. By using steganography and encryption for secure communication, we can prevent the compromise of our information. As a result, encryption algorithms based on the Rijndael series of algorithms increase the security of an effective audio encryption method. This work presents a check-by-bruce-force attack using Rijndael-based algorithms for phonetic encryption. We have successfully established a reliable and efficient use of the Rijndael-based method to secure audio data transmission or storage by carefully evaluating all performance and security considerations. The results of these investigations have helped develop speech encryption methods in various industries and have contributed to voice security. Considering that most cryptography using the Rijndael algorithm only works with text. However, we tried the audio experiment and were able to encrypt multiple blocks with different sizes and keys (128-bit, 192-bit, or 256-bit), especially at 256 blocks and keys, which makes it more difficult to crack. On the other hand, we evaluated the operation of this algorithm, which provides evidence for improving the security and effectiveness of audio encryption methods. We recommend experimenting and evaluating different algorithms and methodologies in all areas of audio and all available media.

Nomenclature

Symbol	Description	Symbol	Description
AES	Advanced Encryption Standard	WAVE	Waveform audio format
DES	Data Encryption Standard	Inv Mix Columns	Inverse Mixed Columns transformation
Inv Shift Rows	The inverse of Shift Rows transformation	RIFF	Resource Interchange File Format
Inv Sub Bytes	Inverse Sub Byte transformation	FIPS	Federal Information Processing Standard
LSB	Least Significant bit	k(i, j)	Value of the key, byte
MixColumns	Mix Column transformation	$r(\alpha, \beta)$	Original value, bit 0-255
NIST	National Institute of Standards and Technology	$r'^{(\alpha,\beta)}$	a new value, bit 0-255
S-Boxes	Substitution Boxes	n_e	multiplying 1 by a times
ShiftRows	Shift Rows transformation	α^{n_e}	Exponentiation, bit 0-255
SubByte	Sub Byte transformation	(x)	fixed polynomial, byte
Triple DES	3-Data Encryption Standard	µ	The multiplied factor by the outcome, byte /or Multiply the result with a fixed polynomial, byte /or.

α	Rows are moved to the left by a certain number of bytes.	**δ**	second-row number
β	row number	**γ**	third-row number

Acknowledgements

The authors are grateful to the University of Baghdad, which allowed us to conduct this research.

Credit Authorship Contribution Statement

Sajaa G. Mohammed: Writing, review, and original draft, validation, Methodology.
Nuhad Salim Al-Mothafar: Proofreading, editing.

Declaration of Competing Interest

The authors declare that they have no financial or other material conflicts of interest that could be construed as affecting the results or interpretation of their manuscript.

REFERENCES

Abdul-Jabbar, S.S., Abed, A.E., Mohammed, S.G. and Mohammed, F.G., 2023. Fast 128-bit multi-pass stream ciphering method. *Iraqi Journal of Science*, 64(5), pp. 2589-2600. https://doi.org/10.24996/ijs.2023.64.5.40

Alabaichi, A. and Altameemi, A.A., 2022. Steganography encryption secret message in video raster using DNA and chaotic map. *Iraqi Journal of Science*, 63(12), pp. 5534-5548. https://doi.org/10.24996/ijs.2022.63.12.38

Albahrani, E.A., Alshekly, T.K. and Lafta, S.H., 2022. A review on audio encryption algorithms using chaos maps-based techniques. *Journal of Cyber Security and Mobility*, 11(1), pp.53-82. https://doi.org/10.13052/jcsm2245-1439.1113

Ali N.H.M., Rahma A.M.S ,2015. An Improved AES encryption of audio wave files, THESIS P.H.D, Department of Computer Science, University of Technology, pp.22-24.

Al-Yasiri, H.A., 2004. A new protocol to design cellular systems with variable spreading factors. *Journal of Engineering*, 10(1), pp.1-12. https://doi.org/10.31026/j.eng.2004.01.01

Aslam, M. and Alkhaldi, A.H., 2015. A novel method of audio steganography using advanced encryption standard. *Nonlinear Engineering*, 4(3), pp. 155-159. https://doi.org/10.1515/nleng-2015-0018

Cheroiu, D.G., Răducanu, M. and Nitu, C.M., 2022, June. Fast image encryption algorithm based on multiple chaotic maps. In *2022 14th International Conference on Communications (COMM)* (pp. 1-4). IEEE. https://doi.org/10.1109/COMM54429.2022.9817317

Daemen, J., Rijmen, V., 1998. The block cipher Rijndael, In *International Conference on Smart Card Research and Advanced Applications* pp. 277–284 .

Easttom, W., 2021. Cryptographic hashes. *Modern Cryptography: Applied Mathematics for Encryption and Information Security*, pp.205-224. https://doi.org/10.1007/978-3-030-63115-4

El Hanouti, I. and El Fadili, H., 2021. Security analysis of an audio data encryption scheme based on key chaining and DNA encoding. *Multimedia Tools and Applications*, 80(8), pp.12077-12099. https://doi.org/10.1007/s11042-020-10153-8

ElShafee, A., 2013. A 64-bit rotor-enhanced block cipher (REBC3). *International Journal of Network Security & Its Applications*, 5(2), p.77. https://doi.org/10.5121/ijnsa.2013.5206

Esttaifan, B.A., 2023. A modified Vigenère Cipher based on time and Biometrics features. *Journal of Engineering*, 29(6), pp. 128-139. https://doi.org/10.31026/j.eng.2023.06.10

Harba, E.S.I., 2018. Advanced password authentication protection by hybrid cryptography & audio steganography. *Iraqi Journal of Science*, 59(1C), pp.600-606. https://doi.org/10.24996/ijs.2018.59.1C.17

Hassan, N.A., Al-Mukhtar, F.S. and Ali, E.H., 2020, November. Encrypt audio file using speech audio file as a key. In IOP Conference Series: *Materials Science and Engineering*. 928(3), p. 032066. IOP Publishing. https://doi.org/10.1088/1757-899X/928/3/032066

Joan, D. and Vincent, R., 2002. The design of Rijndael: AES advanced encryption standard. Information Security and Cryptography, 196.

Kakde, Y., Gonnade, P. and Dahiwale, P., 2015, March. Audio-video steganography. In *2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, 2(6) , pp. 1-6. IEEE. https://doi.org/10.1109/ICIIECS.2015.7192885

Kedem, G. and Ishihara, Y., 1999. Brute force attack on UNIX passwords with SIMD computer. In *8th USENIX Security Symposium USENIX Security* 99. 8(1), pp. 8–18.

Kordov, K., 2019. A novel audio encryption algorithm with permutation-substitution architecture. *Electronics*, 8(5), p. 530. https://doi.org/10.3390/electronics8050530

Li, K., Li, H. and Mund, G., 2023. A reconfigurable and compact subpipelined architecture for AES encryption and decryption. *EURASIP Journal on Advances in Signal Processing*, 2023(1), p.5. https://doi.org/10.1186/s13634-022-00963-3

Mahmood, N.Q., Tahir, Y.F., Hikmat, M., Abdulsatar, M.S. and Baumli, P., 2024. Experimental investigation of the surface roughness for Aluminum Alloy AA6061 in milling operation by taguchi method with the ANOVA technique. *Journal of Engineering*, 30(03), pp.1-14. https://doi.org/10.31026/j.eng.2024.03.01

Majeed, A.H. and Mohammed, S.G., 2017, Using Crypto analysis policies and techniques to create strong password. *Science International (Lahore)*, 29(6), pp. 1297-1308.

Masure, L. and Strullu, R., 2023. Side-channel analysis against ANSSI's protected AES implementation on ARM: end-to-end attacks with multi-task learning. *Journal of Cryptographic Engineering*, 13(2), pp.129-147. https://doi.org/10.1007/s13389-023-00311-7

Mohammed, F.G., Athab, S.D. and Mohammed, S.G., 2021, December. Disc damage likelihood scale recognition for Glaucoma detection. *In Journal of Physics: Conference Series*, 2114(1), p. 012005. IOP Publishing. https://doi.org/10.1088/1742-6596/2114/1/012005

Mohammed, S.G. and Majeed, A.H., 2017. Efficient plain password cryptanalysis techniques. *Iraqi Journal of Science*, 58(A4), pp. 1946-1954. https://doi.org/10.24996/ijs.2017.58.4A.16

Mohammed, S.G., Abdul-Jabbar, S.S. and Mohammed, F.G., 2021, December. Art image compression based on lossless LZW hashing ciphering algorithm. In *Journal of Physics: Conference Series* , 2114(1), p. 012080. IOP Publishing. https://doi.org/10.1088/1742-6596/2114/1/012080

Mustafa, M., Mahmoud, M., Tagelsir, H. and Elshoush, I., 2018, September. A novel enhanced LSB algorithm for high secure audio steganography. In *2018 10th Computer Science and Electronic Engineering (CEEC)* (pp. 125-130). IEEE. https://doi.org/10.1109/CEEC.2018.8674230.

Nițu, C.M., Răducanu, M. and Cheroiu, D.G., 2023. Fast speech encryption algorithm based on Arnold 3D chaotic system. *In Advanced Topics in Optoelectronics, Microelectronics, and Nanotechnologies XI*, 12493, pp. 592-599. SPIE. https://doi.org/10.1117/12.2643008

Răducanu, M., Cheroiu, D.G. and Nițu, C.M., 2023, July. A Novel comparison between different composite chaotic maps applied on sound encryption. In *2023 46th International Conference on Telecommunications and Signal Processing (TSP)* pp. 225-229. IEEE. https://doi.org/10.1109/TSP59544.2023.10197783

Ramalingam, M., Isa, N.A.M. and Puviarasi, R., 2020. A secured data hiding using affine transformation in video steganography. *Procedia Computer Science*, 171, pp. 1147-1156. https://doi.org/10.1016/j.procs.2020.04.123

Rathod, C. and Gonsai, A., 2021. Performance analysis of AES, Blowfish and Rijndael: cryptographic algorithms for audio. In *Rising Threats in Expert Applications and Solutions: Proceedings of FICR-TEAS 2020*, pp. 203-209. https://doi.org/10.1007/978-981-15-6014-9_24

Sethi, P. and Kapoor, V., 2016. A proposed novel architecture for information hiding in image steganography by using genetic algorithm and cryptography. *Procedia Computer Science*, 87, pp.61-66. https://doi.org/10.1016/j.procs.2016.05.127

Shakya, S. and Lamichhane, S., 2016. Secured crypto stegano data hiding using least significant bit substitution and encryption. Journal of Advanced College of Engineering and Management, 2, pp.105-112. https://doi.org/10.3126/jacem.v2i0.16103

Stinson, D.R., 2021. *Techniques for designing and analyzing algorithms*. Chapman and Hall/CRC. https://doi.org/10.1201/9780429277412

Wang, X. and Su, Y., 2019. An audio encryption algorithm based on DNA coding and chaotic system. *IEEE Access*, 8, pp.9260-9270. https://doi.org/10.1109/ACCESS.2019.2963329.2020.

Yaro, A.S., 2004. Surface treatment effects on the corrosion of reinforced steel in concrete exposed to dry condition. *Journal of Engineering*, 10(1), pp. 25-35. https://doi.org/10.31026/j.eng.2004.01.03

تقييم خوارزميه الريجنديل للتشفير الصوتي بواسطة هجوم القوة الغاشمه

سجا غازي محمد*، نهاد سالم المظفر

قسم الرياضيات، كلية العلوم، جامعة بغداد، بغداد، العراق

الخلاصة

استخدام نقل المعلومات سببا رئيسيا لانتشار قرصنة المعلومات، وخاصة المعلومات الرقمية التي تتضمن معلومات صوتية أو غيرها و من أجل الحفاظ على البيانات. يعتبر تشفير الرسائل الصوتية من أكثر الطرق أماناً لحماية المعلومات نظراً لصعوبة الكشف عنها. تميزت الخوارزمية المستخدمة لدينا بالقوة والفعالية والكفاءة والإمكانيات الأمنية نظرًا لاختلاف أطوال المفاتيح وأحجام الكتل (128 بت، 192 بت أو 256 بت). تعتبرالخوارزميةRijndael أداة فعالة لتشفير الملفات الصوتية حيث حصدت العديد من الجوائز، طريقة خوارزمية Rijndael متوافقة مع العديد من تنسيقات الصوت المختلفة؛ سوف نستخدمه لتشفير ملفات WAVE. الخطوة الأولى هي إنشاء مفاتيح باستخدام الطريقة المقترحة تكون من أطوال المفاتيح وأحجام الكتل (128 بت، 192 بت أو 256 بت)، نقوم بتحويل البيانات الصوتية إلى عينة 256 ب بعد ذلك، سيتم إنشاء نص التشفير بالنسبة لنا. بمجرد تشفير الملف الصوتي، ، سيكون في حوزتناعدد من الملفات الصوتية . وبعد ذلك سوف نستخدم نهجا استباقيا لتقييم نجاح هذه المبادرة من خلال هجوم القوة لغاشمه لكسر الشفره حيث يستخدم تشفير Reijndeal، المصمم لمقاومة هجمات القوة الغاشمة، مفتاحًا قويًا وخوارزمية تشفير مناسبة وأجهزة آمنة وبرامج لمنع هجمات القنوات الجانبية وإحباط محاولات تحليل التشفير. تلعب خوارزمية Rijndael دورًا في هذه الدراسة من خلال إظهار فعالية التشفير؛ كما يقدم منظورًا متطورًا. هدفها الأساسي هو حماية البيانات عبر المناظر الطبيعية الرقمية أثناء عمليات النقل والتخزين والحماية.

الكلمات المفتاحية: ريجنديل، تشفير الصوت، تشفير الكتلة ريجنديل، هجوم القوة الغاشمة.