Secure Systems Design and Technology Development. Veliko Tarnovo, Bulgaria
Proceedings of the International Scientific Conference, 2026. Volume I, 357-361

A "Brute Force" Reduced Attack Implementation on 256 - AES Encrypted Voice Stream in a MATLAB Simulink Environment

Elitsa Petkova
Directorate "C4I Systems Development"
Bulgarian Defence Institute
Sofia, Bulgaria
ellydpetkova@gmail.com

***Abstract* — This article presents the steps and the possibility of implementing a reduced "Brute force" attack on a 256-AES encrypted voice stream in a MATLAB Simulink environment. The focus of the study is not on the computational power and the impossibility of performing 2^{256} calculations, but on the selection of the correct decrypted ciphertext from many candidates. The main idea is that decryption is performed based on signal processing, unlike classical cryptanalysis methods. The analog waveform after decryption and decoding carries information whether the corresponding decryption is a function of the "correct" key or not.**

Keywords*— *AES, attack, Brute force, voice stream, cryptanalysis.

I. INTRODUCTION

The AES (Advances Encryption Standard) block cipher was designed in 1997 [1] and is one of the most widely used cryptographic primitives. AES is built on a substitution-permutation network. It encrypts/decrypts blocks of fixed length 128 bits and uses three key lengths, each of which requires a certain number of rounds:

- 128-bit key - 10 rounds;
- 192-bit key - 12 rounds;
- 256-bit key - 14 rounds;

The functionality of the algorithm is presented in Fig. 1 [2] and includes a function for expanding the key according to its length and a series of functions.

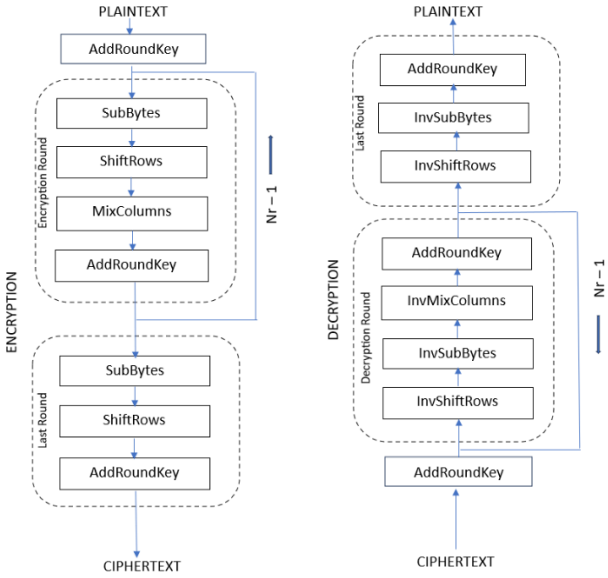

Fig. 1 AES functionality

Online ISSN 3044-7771
https://doi.org/10.68302/std2026.vol1.222

AES is subject to many cryptanalysis. A popular attempt at cryptanalysis of AES whit local collisions is the one applied to the related key model [3]-[6]. It requires not only ciphertexts but also a certain number of corresponding plaintexts, and also has key requirements. From the data complexity perspective, the " Meet in the middle" attack with bicliques, applied to AES [7] is effective, but also has a key requirements. In [8] is presented biclique cryptanalysis of full AES. It also requires ciphertexts and a certain amount of corresponding plaintexts.

From the perspective of the transmitted information, when transmitting voice streams over wireless networks, a potential attacker could only obtain the transmitted ciphertexts.

Therefore, considering the cryptanalysis of voice streams, it can be concluded that the cryptanalytic approaches developed so far, requiring ciphertexts and a certain amount of corresponding plaintexts, are impractical. They only prove certain structural weaknesses in the cipher design. A practical cryptanalytic approach would be one, requires only ciphertexts.

A "Brute force" attack is of no interests to cryptanalysts. Its application requires solving two main problems:

- to provide the necessary computational resources to perform the mathematical calculations;
- to design a method that selects the real decryption from the multitude of such;

To apply a "Brute force" attack on voice streams with 256 bits AES, 2^{256} calculations are required. Now, there is no technology that could provide them. A technological breakthrough and implementation in this direction could be expected in the near future due to the rapid progress of quantum technologies [9] – [11]. On the other hand, the secured computing resource cannot ensure the selection of a real decryption from a multitude of such. In this publication, the focus will be on the implementation and analysis of a method for selecting a real plaintext (decryption) when implementing a "Brute force" attack in a reduced form.

According to the conceptual design, one of the factors for the success of a "Brute force" attack is that it is directly dependent on the output sequences during decryption, determining the form of the analog signal. Fig.2 and Fig. 3 present a conceptual model of encoding, encryption and decryption and decoding of human speech, respectively. The display in Fig.2 visualizes the data before signal processing - encoding and encryption, and the display in Fig.3 - after decoding and decryption. They represent values along the ordinate in the coordinate system. The input data is a function of an audio file that contains a dialogue between two people communicating in Bulgarian. The process of discretization of the signal determines its values along the abscissa. The differences in the values on the two displays, determining the analog signal, are due to the encoding and decoding functions.

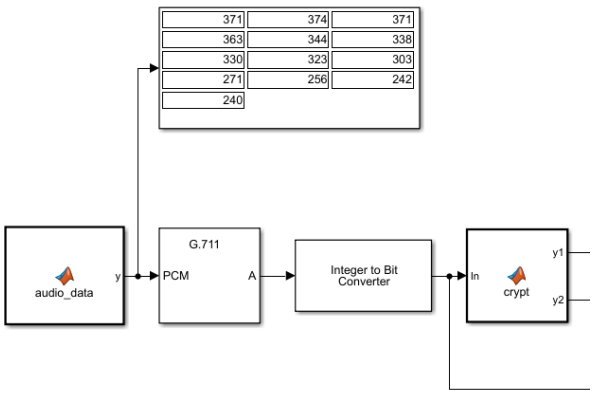

Fig. 2 Conceptual model of encoding and encryption of audio data

The "Verification" block in Fig. 3 performs the corresponding function and outputs a result of "0" if the input data before encryption and after decryption match. Accordingly, it outputs "1" if there is an error in the encryption or decryption module.

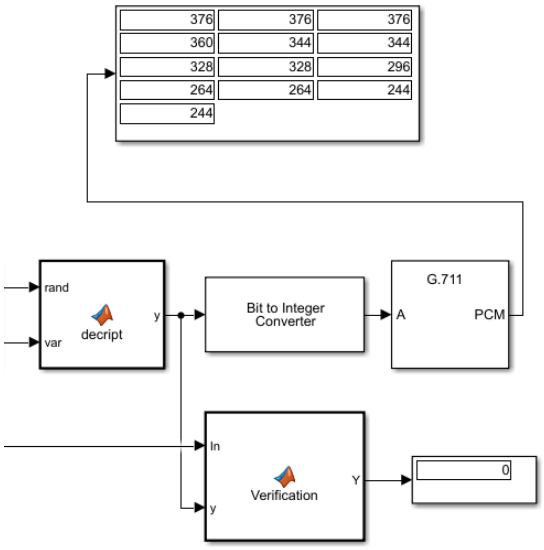

Fig. 3 Conceptual model of decoding and decryption of audio data

This article is structured as follows - in the "TEST SETUP" section, the electrical circuit implementing the reduced "Brute force" attack model in the order of Matlab 2019a – Simulink is described, and in the "EXPERIMENTAL PART", the results of the experiment are described. In "CONCLUSION", the conclusions from the non-standard method of applying a reduced "Brute force" attack are described.

II. TEST SETUP

Many implementations of AES [12], [13], [14], [15] have been created on different development environments. They are not suitable for the present study. For the purpose of the present study, the model should be developed in the Matlab Simulink environment.

The "Brute force" attack model is in reduced form. The model is implemented in a Matlab 2019a – Simulink environment. The computer configuration to be used – CPUx4 (8Tread), 2.2 GHz each core, RAM 8GB DDR4, 2400MHz, operating system – Windows 10 Professional.

Fig. 4 presents some of the blocks necessary for the implementation of the attack. The "audio_data" block contains data in the format of integer values generated from a sound file containing human speech. The "G. 711" codec encodes the data in the PCM (Pulse Code Modulation) to A-low format. Encoded data should be converted to binary, this function is performed by the "Integer to Bit Converter" block. In binary format, they are fed to the encrypt block, which encrypts them with 256 bits AES. The software implementation of the encryption function is presented in [16]. The "Rand" block generates IV (Initialization Vector) and keys required for the attack.

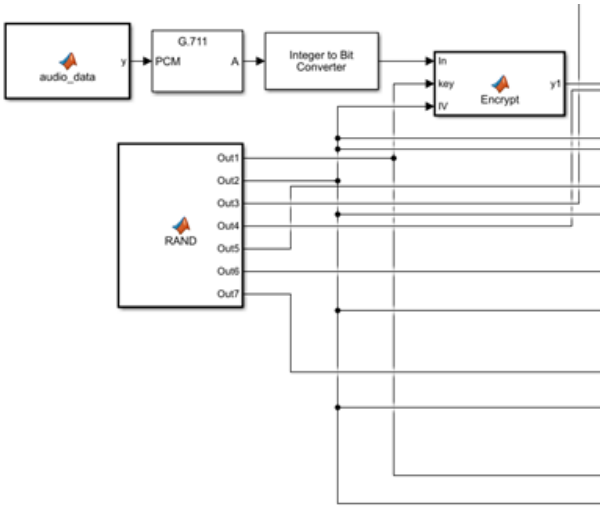

Fig.4 Experimental model of application of Brute Force attack in reduced form, including the blocks – "audio _data", "G. 711", "Integer to Bit Converter", "Rand", "Encrypt";

The attack will be mitigated by not using all 2^{256} key combinations in the decryption process, but only six, five of which are random and one is real. The decryption is performed in parallel and is presented in fig. 5.

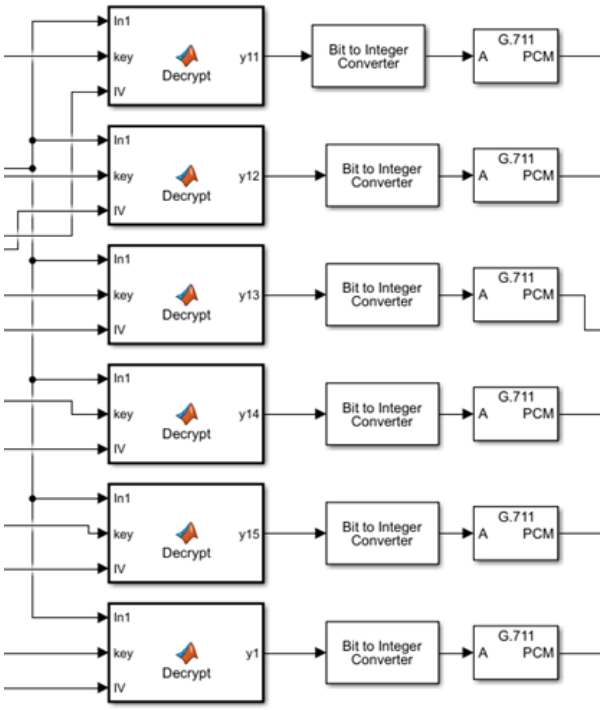

Fig.5 Experimental model of application of Brute Force attack in reduced form, including the blocks – "Decrypt", "Bit to Converter", "G. 711";

The "Bit to Integer Converter" blocks convert the data from binary to integer, after which the "G. 711" codecs decode them, after which they should be sent to the selection block. The result is visualized on a display. They are presented in fig. 6.

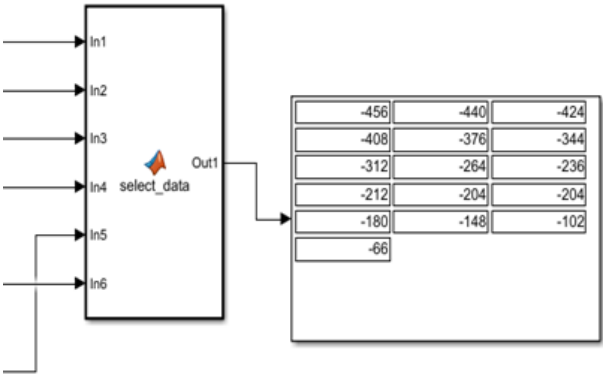

Fig. 6 Experimental model of application of "Brute force" attack in reduced form, including the "select_data" block and display;

III. EXPERIMENTAL PART

An AES encryption consists of 128 bits, equal to 16 integer values. The focus of the conceptual design is that the input sequence, before encoding and encryption, and the output sequence, after decryption and decoding, represent sixteen analog values along the ordinate in the coordinate system.

Since they are a function of human speech, they change smoothly. When decrypted with the wrong key, a sequence of randomly changing integer values is obtained, shown in Fig. 7. This property is the basis for implementing a method for selecting a decrypted sequence with the correct key.

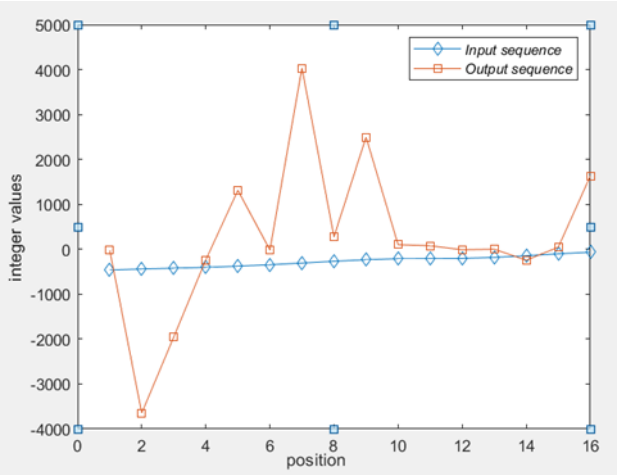

Fig. 7 Input sequence and output decrypted with the wrong key

The focus when creating a selection algorithm is to look for dependencies in the output sequences when decrypting with a "correct" key, which are absent when decrypting with a "wrong" one. This is done by searching for the distance between each two adjacent values. When decrypting with a "correct" key, this distance has certain limitations. The selection algorithm is logically connected to a verification method. Their development should be described in a separate publication.

After conducting multiple experiments with different input values and key sequences, it can be concluded that the block performs the function of selecting the correct decryption.

Fig. 8 visualizes the one input and correctly decrypted output sequence.

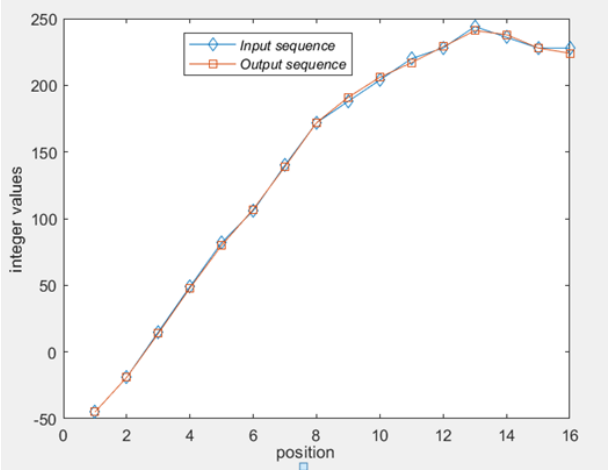

Fig.8 Input and correctly selected output sequence

One of the results of the conducted experimental studies, visualizing all six decryptions, is shown in Fig. 9.

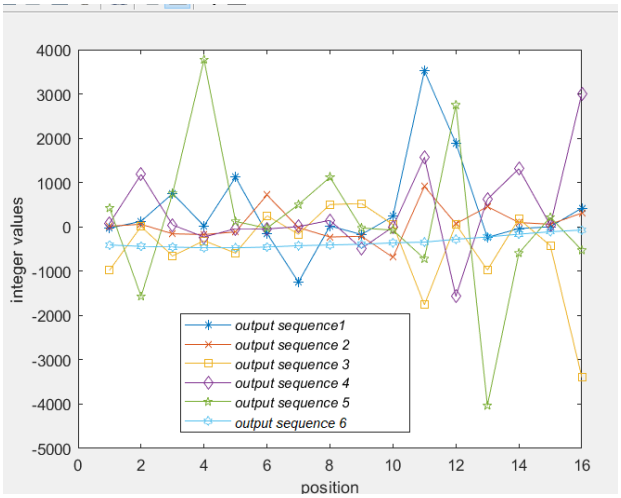

Fig.9 Decrypted output sequences

IV. CONCLUSION

A method for reduced form a "Brute force" attack in Matlab 2019a – Simulink environment has been developed. A selecting method the correct decryption from the set of such has been developed. The method is not based on standard cryptanalytic approaches, but on signal processing. The method has been experimentally studied. The next step in this research is to develop a method for verification of the selected sequence.

REFERENCES

[1] J. Daemen, V. Rijmen, "The Design of Rijndael: AES - The Advanced Encryption Standard", Springer, 2002, Available: https://link.springer.com/book/10.1007/978-3-662-04722-4 [Accessed April 27, 2026]

[2] R. Ganesh, B. Khan, A. Khan, A. Kasin, "A panoramic survey of the advanced encryption standard: from architecture to security analysis, key management, real – world applications, and post – quantum challenges", International Journal of Information, vol. 24, no 216 (2025), Available: https://link.springer.com/article/10.1007/s10207-025-01116-x [Accessed June 9, 2026], https://doi.org/10.1007/s10207-025-01116-x

[3] A. Biryukov, O. Dunkelman, N. Keller, D. Khovratovich, A. Shamir, "Key recovery attacks of practical complexity on AES-256 variants with up to 10 rounds", In EUROCRYPT'10, vol. 6110 of Lecture Notes in Computer Science, pp. 299–319. Springer, 2010.

[4] A. Biryukov, D. Khovratovich, "Related-Key Cryptanalysis of the Full AES-192 and AES-256", In ASIACRYPT'09, vol. 5912 of Lecture Notes in Computer Science, pp. 1–18. Springer, 2009.

[5] A. Biryukov, D. Khovratovich, I. Nikoli´c, "Distinguisher and related-key attack on the full AES-256", In CRYPTO'09, vol. 5677 of Lecture Notes in Computer Science, pp. 231–249. Springer, 2009

[6] A. Biryukov, I. Nikoli´c, "Automatic Search for Related-Key Differential Characteristics in Byte - Oriented Block Ciphers: Application to AES, Camellia, Khazad and Others", In EUROCRYPT'10, vol. 6110 of Lecture Notes in Computer Science, pp. 322–344. Springer, 2010.

[7] Orr Dunkelman and Nathan Keller. "The effects of the omission of last round's MixColumns on AES", Inf. Process. Lett., 110(8-9), pp. 304–308, 2010, Available: https://eprint.iacr.org/2010/041.pdf [Accessed April 27, 2026]

[8] A. Bogdanov, D. Khovratovich, Ch. Rechberger, "Biclique Cryptanalysis of the Full AES", LectureNotes in Computer Science, pp. 344–371. Springer, 2011, Available: https://www.researchgate.net/publication/221326929_Biclique_Cryptanalysis_of_the_Full_AES [Accessed April 27, 2026]

[9] L. Grover, "From Schrödinger's equation to the quantum search algorithm", American Journal of Physics pp. 769-777, 2001. Available: https://arxiv.org/pdf/quant-ph/0109116, [Accessed June 8, 2026]

[10] C. H. Bennett, E. Bernstein, G. Brassard, U. Varirani, "The strength and weaknesses of quantum computation". SIAM Journal on Computing, 26(5), pp. 1510-1523, 1996. Available: https://arxiv.org/pdf/quant-ph/9701001, [Accessed June 8, 2026]

[11] S. Rao, D. Mahto, D. Yadav, Danish, D. Khan., "The AES-256 Cryptosystem Resist Quantum Attacks", International Journal of Advanced Research in Computer Science, 2017, 8(3) p. 404-408 Available:https://www.researchgate.net/publication/316284124_The_AES-256_Cryptosystem_Resists_Quantum_Attacks,[Accessed June 8, 2026]

[12] R. Andriani, S. Wijayanti, F Wibowo, "Comparision of AES 128, 192 and 256 Bit Algorithm For Encryption and Descryption File", 3rd International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE), Yogyakarta, Indonesia, 2018.

[13] M. Muzakkir, M. Nadzri, A. Ahmad, A. Amira, "Implementation of Advanced Encryption Standard (AES) for Wireless Image Transmission using LabVIEW", IEEE 16th Student Conference on Research and Development (SCORed), Bangi, Malaysia, 2018.

[14] Ch. Su, T. Lin, Ch. Huang, Ch. Wu, "A High – Throughput Low – Cost AES Processor" IEEE Communications Magazine, vol. 41, p. 86-91, Available: https://ieeexplore.ieee.org/document/1252803, http://www.umi.com/proquest/ [Accessed June 9, 2026], https://doi.org/10.1109/MCOM.2003.1252803

[15] A. Kumar, N. Tiwar, "Effective Implementation and Avalanche Effect of AES", International Journal of Security, Privacy and Trust Management (IJSPTM), vol. 1, no. 3/4, p. 31-35 Available: https://www.researchgate.net/publication/267802536_Effective_Implementation_and_Avalanche_Effect_of_AES, [Accessed June 9, 2026]

[16] E. Petkova, M. Angelov, „Implementation of "Advanced Encryption Standard" in the Matlab – Simulink environment" Scientific conference "Current security problems". October 9-10, 2025, Veliko Tarnovo, Bulgaria , 2025