Comparison of Cryptography by Chaotic Neural Network and by AES

Lenka Skovajsová
Institute of Informatics
Slovak Academy of Sciences
Bratislava, Slovakia
lenka.skovajsova@savba.sk

Abstract—**In this paper, the two methods for ciphering are presented and compared. The aim is to reveal the suitability of chaotic neural network approach to ciphering compared to AES cipher. The durations in seconds of both methods are presented and the two methods are compared. The results show, that the chaotic neural network is fast, suitable for ciphering of short plaintexts. AES ciphering is suitable for longer plaintexts or images and is also more reliable.**

Keywords—cryptography, chaotic neural networks, AES cipher, chaotic maps, text encryption

I. Introduction

Cryptography is very important research area in these times. As the attacks on the ciphered messages (text, images, ...) are very sophisticated, it is required for the ciphering algorithms to be more complex and robust to resist to these attacks.

One such robust ciphering method is Advanced Encryption Standard (AES) [9, 10, 11, 12, 13, 14], which is so complex that it can resist to majority of the attacks on it. AES is the cipher that was elected out of 15 other candidate ciphers in the year 2000 because it proved to resist against majority attacks on it. AES is considered as the traditional state-of-the-art encryption technique.

Another area of creating ciphers is by chaotic neural networks that use chaotic maps for generating pseudo-random sequence of integers for construction. Chaotic neural network for cryptography that uses pseudo-random sequence created by the chaotic map firstly emerged in the beginning of the 21st century [6, 7, 8]. Chaotic neural network is very sensitive to its input parameters that create a key, so only little change of these parameters can cause the chaotic behavior and can make impossible to restore the original message from the ciphered message.

II. AES Cipher

The Advanced Encryption Standard (AES) cipher belongs to Rijndael ciphers developed by two Belgian cryptographers, Joan Daemen, and Vincent Rijmen [10]. Rijndael is a family of ciphers that groups ciphers of different key sizes and block sizes. It was elected out of fifteen candidates for Advanced Encryption Standard in 2. October 2000.

The used AES cipher is described in detail in [9]. Here is only a brief sketch.

AES is a block cipher. Used variant of AES for ciphering firstly divides the plaintext message into 16-byte blocks and these blocks are rearranged to the 4x4 matrix.

AES uses ciphering keys of three sizes, 128, 192, and 256 bits. The AES ciphering algorithm comprises of four basic steps:

The very first step (Step 0) is addroundkey which is also the last step. In addroundkey, each byte of the state will be combined with a round key by using a bitwise xor.

Step 1: Byte substitution – each byte is substituted by a byte from a lookup table.

Step 2: Shifting rows – last three rows are shifted by a certain number of positions.

Step 3: Mixing the columns – combines the four bytes in the specific column.

Step 4: Adding a round key to the state.

These rounds (Steps 1-4) are repeated depending on the size of the key. For key size of 128, are these steps repeated 10 times, for 192 key size 12 times, and for 256 key size 14 times. Used AES cipher uses the key of size 128, so it repeats these steps 10 times, one for each round.

Deciphering by AES algorithm is similar:

Step 1: Inverse byte substitution

Step 2: Inverse of shifting rows

Step 3: Inverse of mixing columns

Step 4: Addition of round key to the state

Deciphering is also repeated based on the size of the key.

AES uses the same key for ciphering and for deciphering, so it is symmetric cipher.

Ciphering and deciphering is depicted in Fig. 2 and Fig. 4 and applying round key is shown in Fig. 1 and Fig. 3.

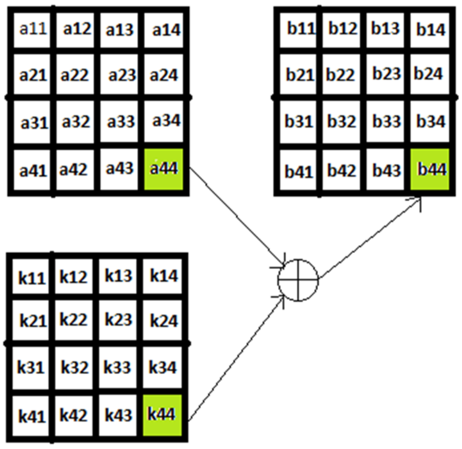

Fig. 1. AES addroundkey operation

978-1-7281-5625-5/19/$31.00 ©2019 IEEE

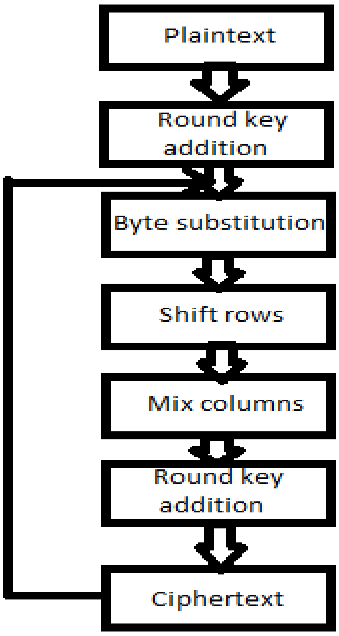

Fig. 2. AES ciphering

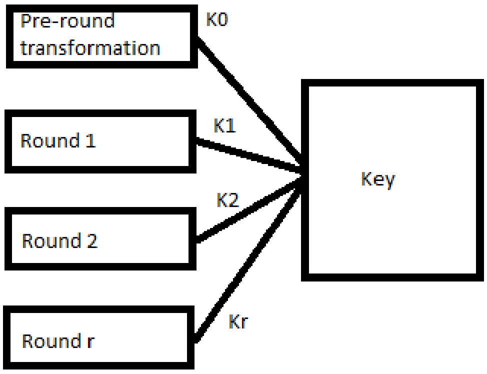

Fig. 3. Round key addition

III. Chaotic Neural Network

Chaotic neural network is a kind of neural network (for example Hopfield network) that shows chaotic behavior [2]. Chaotic behavior of Hindmar Rose neurons is described in [1]. In [3], the Hopfield neural network with switching chaotic maps is used for image cryptography. In [5], the neural network is used to encrypt jpeg images.

It is shown, that chaotic neural networks show chaotic behavior, when the parameters are chosen suitably. In [4], the delayed neural networks are used to obtain chaotic behavior.

The used chaotic neural network for cryptography for comparison with AES is described in detail in [8], here we will mention only brief sketch. In [6, 7], similar chaotic ciphers are described based on perceptron model.

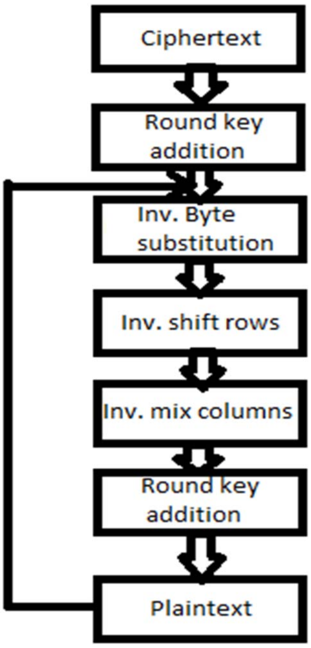

Fig. 4. AES deciphering

Used neural network [8] is represented by two equal networks, the coder and the decoder. Coder is used for ciphering and decoder for deciphering. The length of ciphertext is the same as the length of the original text.

The coder of this scheme is same as the decoder, so the used ciphering is symmetric.

The coder at first computes the chaotic sequence $x(t)$ of the same length as the ciphered text by using some type of chaotic map. Then, this chaotic sequence is used to determine the weights W and biases θ. Then, the ciphertext is computed on the base of these parameters.

$$C(t) = f(WP(t) + \theta)$$

where $P(t)$ is the plaintext and $C(t)$ is the ciphered text, and $f(x)$ is a function that takes value 1, if $x \geq 0$, and is 0, otherwise.

Used ciphering algorithm can be briefly described in the following four steps:

Step 1: Determine the parameters μ, and $x(0)$.

Step 2: On the base of these parameters, determine the chaotic sequence $x(t)$, and $b(t)$, $t = 1{:}\,length(plaintext)$ using chaotic map.

Step 3: On the base of these settings create the neural network - coder and decoder.

Step 4: Cipher the plaintext by using created coder. (Decipher the ciphered text by this neural network).

For creating chaotic sequence part in time t, $x(t)$, all previous inputs $x(1) \ldots x(t-1)$ are used. That is the reason, why long ciphers are ciphered (deciphered) extremely slowly.

IV. Experiments

For testing processing times, we used following hardware: 8 processors and 16GB RAM.

For chaotic neural networks we used the following software: MS Windows 10, 64bit, MATLAB 7, 64bit. For AES we used following software: Windows 10, 64bit, MATLAB 7, 64bit, MATLAB source code used in [5].

For our experiments we used 30 randomly generated files (by file generator). Five of them are of size 512b, five of size 1024b, five of size 2048b, five of size 3KB, five of size 30KB, and five of size 3MB.

We made experiments on each of the files three times for AES and three times for chaotic neural networks and measured the time of ciphering and the time of deciphering separately. For five different files of same size we made 15 (5x3) experiments for ciphering by AES, 15 experiments for deciphering by AES, 15 experiments for ciphering by chaotic neural network, and 15 experiments for deciphering by chaotic neural network. The result for each 15 experiment groups are averaged and their results are shown in Table 1 and graphically shown in Figs 5 and 6. Note that in the graphical representations there is not shown the last case of experiments, which is for the 3MB files, due to space limitations and due to fact that there is not precise result for the CNN.

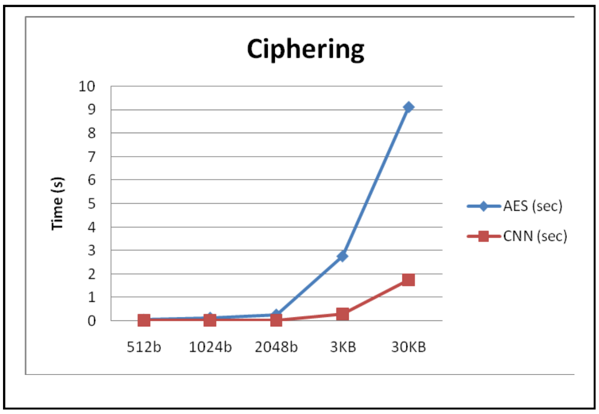

Fig. 5. Comparison of ciphering by AES and CNN

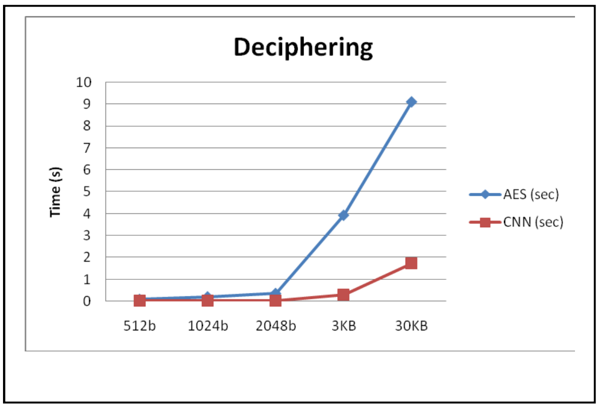

Fig. 6. Comparison of deciphering by AES and CNN

TABLE I. Comparing of AES and CNN performance

Ciphering	**File size**		
	512b	***1024b***	***2048b***
AES (sec)	0,061373	0,14478	0,278153
CNN (sec)	0,037587	0,04264	0,049693
Ciphering	***3KB***	***30KB***	***3MB***
AES (sec)	2,76693	9,12209	2779,6
CNN (sec)	0,3116	1,74542	>5400

Deciphering	**File size**		
	512b	***1024b***	***2048b***
AES (sec)	0,09994	0,214973	0,36888
CNN (sec)	0,032453	0,03668	0,045353
Deciphering	3KB	30KB	3MB
AES (sec)	3,940763	9,12209	3976,7
CNN (sec)	0,31208	1,729927	>5400

V. Conclusions

From the experiments we can see that AES is slower both in ciphering and deciphering than chaotic neural network. But it should be marked that used chaotic neural network uses a 1D chaotic map, which depends only on two parameters, what makes it more easily breakable than AES. AES is standard that is robust against many different attacks what can be seen from different works such as [12, 13, 14]. So, used chaotic map cipher is suitable for ciphering short texts, AES is more suitable for ciphering large plaintext or images due to its shorter time of ciphering large files.

One disadvantage of chaotic neural networks is that they become after certain size of plaintext periodic, so they are more likely to be broken for large files. This is mainly for chaotic neural networks that use 1D chaotic maps where generated chaotic sequence has lower period. 2D and 3D chaotic maps are used for encryption more often than 1D chaotic maps.

Both algorithms created the ciphertexts of the same size as the original plaintexts and both ciphers are symmetric.

Acknowledgment

This work was supported by Slovak Scientific Grant Agency VEGA No. 2/0167/16.

References

[1] HANSEL, David; SOMPOLINSKY, Haim. Synchronization and computation in a chaotic neural network. *Physical Review Letters*, 1992, 68.5: 718.

[2] AIHARA, Kazuyuki; TAKABE, T.; TOYODA, M. Chaotic neural networks. *Physics letters A*, 1990, 144.6-7: 333-340.

[3] YU, Wenwu; CAO, Jinde. Cryptography based on delayed chaotic neural networks. *Physics Letters A*, 2006, 356.4-5: 333-338.

[4] ZHANG, Huaguang, et al. Adaptive synchronization between two different chaotic neural networks with time delay. *IEEE Transactions on Neural Networks*, 2007, 18.6: 1841-1845.

[5] LIAN, Shiguo, et al. A chaotic-neural-network-based encryption algorithm for JPEG2000 encoded images. In: *International Symposium on Neural Networks*. Springer, Berlin, Heidelberg, 2004. p. 627-632.

[6] LIAN, Shiguo, et al. A chaotic-neural-network-based encryption algorithm for JPEG2000 encoded images. In: *International Symposium on Neural Networks*. Springer, Berlin, Heidelberg, 2004. p. 627-632.

[7] WANG, Xing-Yuan, et al. A chaotic image encryption algorithm based on perceptron model. *Nonlinear Dynamics*, 2010, 62.3: 615-621.

[8] SU, Scott; LIN, Alvin; YEN, Jui-Cheng. Design and realization of a new chaotic neural encryption/decryption network. In: IEEE APCCAS 2000. 2000 IEEE Asia-Pacific Conference on Circuits and Systems. Electronic Communication Systems.(Cat. No. 00EX394). IEEE, 2000. p. 335-338.

[9] Kumar, D.Lohit & Reddy, AR & S A K, Jilani. (2016). Implementation of 128-bit AES algorithm in MATLAB. International Journal of Engineering Trends and Technology (IJETT). 33. 126. 10.14445/22315381/IJETT-V33P223.

[10] DAEMEN, Joan; RIJMEN, Vincent. *The design of Rijndael: AES-the advanced encryption standard*. Springer Science & Business Media, 2013.

[11] RIJMEN, Vincent; DAEMEN, Joan. Advanced encryption standard. *Proceedings of Federal Information Processing Standards Publications, National Institute of Standards and Technology*, 2001, 19-22.

[12] GRASSI, Lorenzo; RECHBERGER, Christian; RØNJOM, Sondre. Subspace trail cryptanalysis and its applications to AES. *IACR Transactions on Symmetric Cryptology*, 2016, 192-225.

[13] TAO, Biaoshuai; WU, Hongjun. Improving the biclique cryptanalysis of AES. In: *Australasian Conference on Information Security and Privacy*. Springer, Cham, 2015. p. 39-56.

[14] ZHAO, Kaixin; CUI, Jie; XIE, Zhiqiang. Algebraic cryptanalysis scheme of AES-256 using Gröbner basis. *Journal of Electrical and Computer Engineering*, 2017, 201.