A Comparative Performance Analysis of Modified Dömösi's Cryptosystem and Data Encryption Standard

G. Khaleel, S. Turaev, M.I.M. Tamrin and I.F. Al-Shaikhli
Kulliyyah of Information and Communication Technology, International Islamic University Malaysia
53100 Kuala Lumpur, Malaysia, ghassan.khaleel@live.iium.edu.my, {sherzod, izzuddin, imadf}@iium.edu.my

Abstract**—**The performance has a central importance for all cryptosystems. This paper aims to provide a framework and platform to evaluate a performance of Modified Dömösi's Cryptosystem, a stream cipher based on finite automata, comparing to Data Encryption Standard. The performance is evaluated in terms of encrypting time and throughput.

Keywords**—** Dömösi's cryptosystem, Data Encryption Standard, deterministic finite automata, performance.

I. Introduction

The security of information is one of the most important aspects in communications, which always demands to improve the existing cryptosystems, and design new ones with high security and performance in real-time applications due to enhancing the danger of hacking effort. Therefore, many cryptosystems have been developed to improve the security of information. A cryptosystem encompasses the principles, and method of transforming an intelligible message (plaintext) into another one that is unintelligible (ciphertext) and then retransforming that message back to its original form.

Generally, cryptographic systems can be classified into two main types: symmetric-key and asymmetric-key systems. Symmetric-key systems use a single key that both sender and recipient have, whereas public-key systems use two keys: a public key known to everyone and a private key that only the recipient of the messages uses. Symmetric-key systems can be also broadly classified into: block ciphers and stream ciphers. The basic idea of a block cipher is to break a plaintext into fixed large length blocks, and encrypt each independently through many rounds of confusions and diffusions. While a stream cipher encrypts a sequence of data, typically a bit or byte, but uses a key or sequence of keys that generated by a key generator to encrypt blocks.

II. Data Encryption Standard

Data Encryption Standard (DES) [1,2] is a symmetric block cipher was introduced by IBM, and approved as a federal standard in November 1976, and published on 15 January 1977. Based on NIST, DES cipher is considered to have a catalyst for the academic study of cryptography, particularly of methods to crack block ciphers and its performance [3]. After twenty years, with many attacks, techniques and methods recorded the weaknesses of DES, which resulted to break DES [4,5].

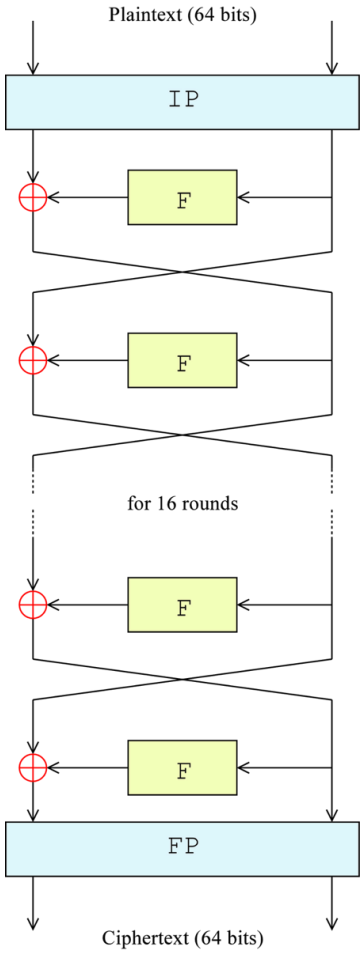

Fig. 1 DES structure

Typically, DES operates with block size 64 bit of plaintext, and uses 64-bit key length, where, 8-bits are used for checking the parity of transformation. Hence, the effective key length is 56-bits long. As shown in Figure 1, the algorithm's structure consists of 16 rounds with initial permutation (IP) and final permutation (FP). In the encryption algorithm, the block of plaintext divided into two 32 bits, and each one of them processed alternately by Feistel f-function. The f-function scrambles half a block together with a part of the key. The output from the f-function is combined using XOR operation with the other half of the block, and the halves are swapped before the next round as shown in Figure 2.

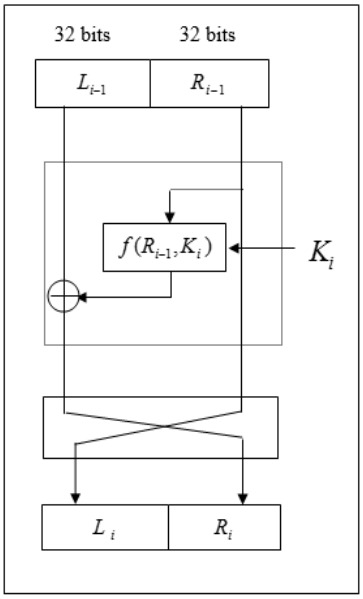

Fig. 2 A round in DES

After the final round, the halves are swapped. Where, the Feistel function consists of 16 rounds of permutations and substitutions. In the decryption mode, same processes are used, the only difference in the sub keys are applied in reverse order.

III. Modified Dömösi's Cryptosystem

In 2008, Dömösi [6,7] introduced a symmetric cryptosystem base on Rabin-Scott automata i.e., deterministic finite automata without outputs, which act as keys for encrypting plaintexts and decrypting ciphertexts. In this way, Dömösi's cryptosystem is similar to Mealy machine based cryptosystems: the encoding and decoding are performed using a key automaton, but it is different from Mealy machine in generating ciphertext: it does not generate the ciphertext by combining the plaintext bit stream with a random bit stream (key) using the exclusive or operator (XOR). Dömösi's cryptosystem also differs from cellular automata based cryptosystems in producing the ciphertext. But it is similar to cellular automata in that the key automaton based on finite automaton without outputs and its random number generator is independent of the key. In this matter, Dömösi's cryptosystem uses any random number generator which is proved to be random indeed. For instance, it can use any true random number generator sources, any radioactive generator or any software can generate the random number.

On the other hand, Dömösi's cryptosystem has many advantages over many others stream ciphers. Firstly, the random number generator is independent from the key. Secondly, this system cannot be attacked by methods used for defeating cryptosystems not based on finite automata. In addition to these advantages, it can be implemented the cryptosystem in the software and hardware efficiently due to the simplicity of the operations used.

Moreover, Dömösi's cryptosystem overcomes some complicated mechanisms in broadcasting/datacasting systems, i.e., this cryptosystem makes frequent key changes unnecessary. However, this cryptosystem suffers from some practical difficulties in construction of ciphertexts: the algorithm has exponentially many backtrackings in the worst-case to construct a ciphertext blocks for each plaintext symbol, which affect the entire performance of the cryptosystem.

Further improvements and extensions of Dömösi's cryptosystem can be found in [8-14]. To overcome the backtracking drawback, and improve the performance of Dömösi's cryptosystem, G. Khaleel et. al. [8,9] proposed an additional control system used together with the second Dömösi's encryption algorithm without affecting the security of the original cryptosystems, i.e., the proposed cryptosystem is as secure as Dömösi's cryptosystem. This control system prevents backtracking in the algorithm by generating the "control" vectors.

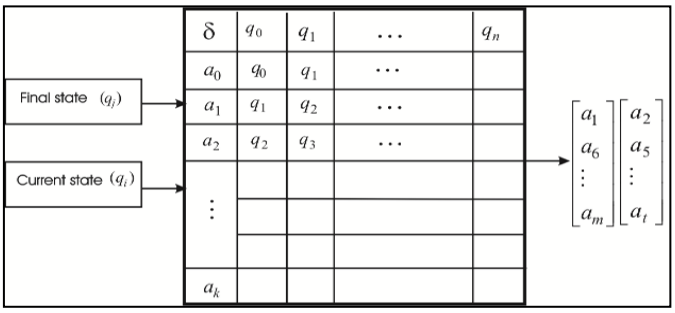

Fig. 3 Initialization Stage

The trade-off between a constant amount of space and time of the algorithm allows constructing a sequence of transitions which takes the automaton from the current state to one of the target final states without intermediate final states and backtracking in a linear time. The proposed control system consists of control vectors of nonfinal and final states constructed with respect to a state, the input signals and the final states, where the function of the control system can be divided into two stages: the initialization and operation stage.

In the initialization stage, for a key automaton $\mathcal{A} = (\mathrm{Q}, \Sigma, \delta, \mathrm{q}_0, \mathcal{F})$ where Q is a finite set of states, Σ is an alphabet, δ is a transition function, q_0 is the initial state,

and $\mathcal{F}$ is a set of final states, the control system generates two control vectors V_1 and V_2 in advance only once (see Figure 3), i.e., before the encoded message is generated, the cipher must initialize the control vectors (V_1, V_2) where V_1 consists of all input signals ($a_i \in \Sigma$) that take the automaton from the current state ($p \in Q$) to any non-final state, whereas V_2 consists of all input signals ($a_j \in \Sigma$) that take the automaton from the current state p to one of the target final states ($q \in \mathcal{F}$).

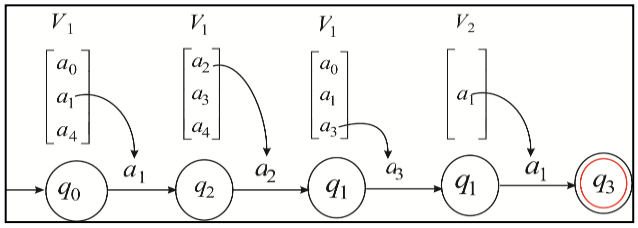

Fig. 4 Operation Stage

In the operation stage as shown in Figure 4, the key automaton $\mathcal{A} = (Q, \Sigma, \delta, q_0, \mathcal{F})$ generates a ciphertext block $w_i \in \Sigma^+$ of length t for the plaintext symbol $i \in \Pi$,

where t is randomly selected from $[s_{min}, s_{max}]$ and V is either the vector V_1 or the vector V_2. First, the algorithm constructs a prefix, w_i, of length $t-1$ by randomly selecting signals from vectors V_1. Second, it selects a random signal from V_2 finalizing the construction of the block cipher w_i.

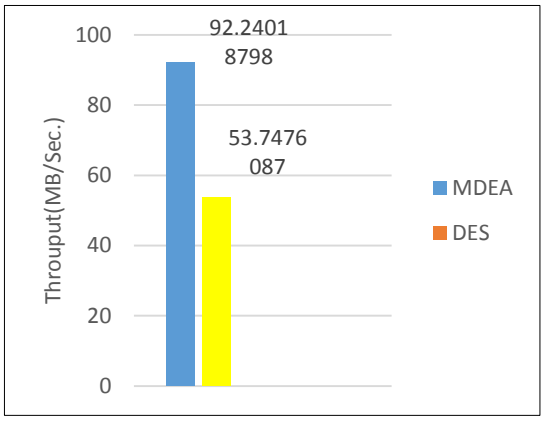

Fig. 5 Encryption time of MDEA and DES

IV. Performance Analysis

In this section, a comparative analysis in terms of encrypting time and throughput of modified Dömösi's encryption algorithm (MDEA) and Data Encryption Standard (DES) is provided. For modified Dömösi's encryption algorithm (MDEA) we use large key automaton $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$, where $|Q| = 256$, $|\Sigma| = 256$, $|F| = 16$, $s_{min} = 5$ and $s_{min} = 5$.

To estimate the performance of DES, we use DES algorithm based on the crypto++ standard library version 5.6 [10].

The speed measurement was performed on Lenovo Notebook E430 having Intel(R) Core(TM) i5-3230M CPU 2.6 GHz with 4 GB RAM under 64-bit Operating System Windows 10. The simulation programs are written in C++ under Visual studio 2013.

Finally, the results are shown in Figure 5. Here, we compare the encryption time of DES and MDEA algorithms over different plaintext sizes, ranges from 16KB to 1MB. MDEA takes less time to produce the ciphertext blocks with respect to DES. Whereas, in terms of ciphertext throughput, Figure 6 shows DES has less throughput when compared with MDEA, the throughput of DES reach to approximately 57% of the throughput of MDEA.

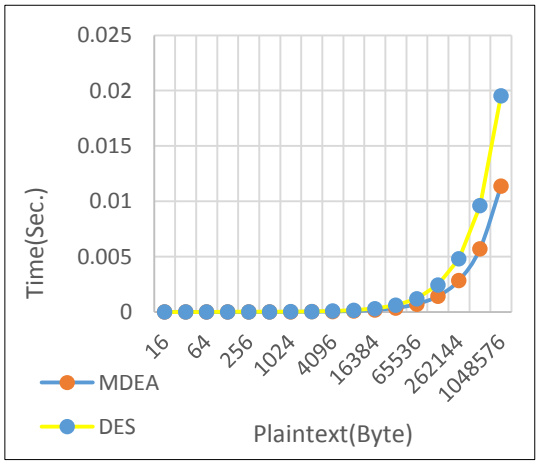

Fig. 6 Ciphertext throughput of MDEA and DES

V. Conclusions

This paper conducted a performance evaluation of modified Dömösi's encryption algorithm (MDEA) and Data Encryption Standard (DES). The simulation results showed that DES cipher needs to more processing time to encode a message. While in terms of throughput, the MDEA has high ciphertext throughput compared to DES. Moreover, we described the structure of mentioned cryptosystems.

VI. Acknowledgements

This work has been supported through International Islamic University Malaysia Research Initiative Grant Scheme **RIGS16-368-0532**.

References

[1] C. Paar and J. Pelzl, "Understanding Cryptography: A Textbook for Students and Practitioners", Springer, 2009.

[2] B. Schneier, "Applied Cryptography: Protocols, Algorithms, and Source Code in C", John Wiley & Sons, Inc., 1996.

[3] Data Encryption Standard, a. https://en.wikipedia.org /wiki/Data_Encryption_Standard.
[4] E. Biham and A. Shamir, "Differential Cryptanalysis of DES-like Cryptosystems", Journal of Cryptology, 4(1), pp. 3–72, 1991.
[5] D. Coppersmith, "The Data Encryption Standard (DES) and Its Strengths against Attacks", IBM Journal of Research and Development 38 (3), pp. 243–50, 1994.
[6] P. Dömösi, "A novel cryptosystem based on finite automata without outputs", In: M. Ito, Y. Koboyashi, and K. Shoji (eds.), Automata, Formal Languages and Algebraic Systems, World Scientific, p. 23–32, 2008.
[7] P. Dömösi, "Apparatus and Method for Protection of Conditional Access Broadcasting and Datacasting", doi: WO 2008/110852 A1, 2008.
[8] G. Khaleel, S. Turaev, M.I. Mohd Tamrin and I.F. Al-Shaikhli, "A Performance Improvement of Dömösi's Cryptosystem", AIP Conference Proceedings 1705, 020007, 2016.
[9] G. Khaleel, S. Turaev, M.I. Mohd Tamrin and I.F. Al-Shaikhli, Performance and Security Improvements of Dömösi's Cryptosystem. International Journal of Applied Mathematics and Statistics, 55(2), 2016, pp. 32–45.
[10] G. Khaleel, S. Turaev and T. Zhukabayeva, "A Novel Stream Cipher Based on Nondeterministic Finite Automata", Information Technologies in Science, Management, Social Sphere and Medicine (ITSMSSM 2016), 23-26 May, 2016, Tomsk, Russia, Atlantis Press, 2016, pp. 110–191.
[11] G. Khaleel, S. Turaev, M.I. Mohd Tamrin and I.F. Al-Shaikhli, "A New Block Cipher Based on Finite Automata Systems", International Journal on Perceptive and Cognitive Computing, 2(1), 2016, pp. 23–26.
[12] M. Shafriezal, S. Razalai, G. Khaleel, and S. Turaev, "A New Symmetric Cryptosystem Based on Permutation Matrices", International Journal on Perceptive and Cognitive Computing 2(2), 2016, pp. 36–40.
[13] G. Khaleel, S. Turaev, M.I. Mohd Tamrin and I.F. Al-Shaikhli, "An Overview of Cryptosystems Based on Finite Automata", Journal of Advanced Review on Scientific Research 27(1), 2016, pp. 1–7.
[14] G. Khaleel, S. Turaev, M.I. Mohd Tamrin and I.F. Al-Shaikhli, "A Symmetric Cryptosystem Based on Nondeterministic Finite Automata", Journal of Theoretical and Applied Information Technology 95(10), 2017, pp. 1489–1498.
[15] "Crypto++ 5.6.0 Benchmarks", https://www.cryptopp.com/.