Journal of Multidisciplinary Engineering Science and Technology (JMEST)
ISSN: 3159-0040
Vol. 2 Issue 12, December - 2015

Comparison between AES, Camellia and SEED

M. A. Mioc
Computer Science Department
Politehnica University of Timisoara
Timisoara, Romania
mmioc@cs.upt.ro

S. G. Pentiuc
Faculty of Electrical Engineering and Computer Science
University "Stefan cel Mare"
Suceava, Romania
pentiuc@eed.usv.ro

***Abstract*—The actual increasing of the networks interconnections produces a growing necessity to keep the information secure from the hackers and eavesdroppers attacks. ISO/IEC 18033 - 3: 2010 specifies the following 128-bit block ciphers: AES, Camellia and SEED. This paper describes a research based on comparing the times required by the mentioned three algorithms in function of the plaintext lengths. The Open SSL library from Ubuntu Linux 14.04 LTS has been used.**

Keywords—security; ciphers; plaintext; algorithm; key size

I. INTRODUCTION

In accordance with current standards of the encryption contained in ISO/IEC 18033-3-2010 the encryption ciphers having the purpose of data confidentiality are presented in the following rows.

ISO/IEC 18033-3:2010 specifies block ciphers. A block cipher is a symmetric encipherment system where the encryption algorithm operates on a block of plaintext to obtain block of cipher text.

A plaintext is a well-defined length string of bits.

The following algorithms are specified:

- 64-bit block ciphers: TDEA, MISTY1, CAST-128, HIGHT;

- 128-bit block ciphers: AES, Camellia, and SEED.

ISO standards are reviewed every five years, so that last time was in 2013.

Starting from the year 2000, when Rijndael became the winner of the international contest of cryptographic algorithms, there have been made several kinds of comparison between existing encryption ciphers.

There have been attempted comparisons on the basis of criteria hard or soft, as a function of time, the size of the encrypted texts, depending on the type of input date lot and more.

However a comparison focused on AES, Camellia and SEED does not exist.

This was the idea and the motivation to achieve this research presented below.

II. RELATED WORK

Over time several kinds of comparisons have been made between algorithms.

These comparisons focus on many evaluation criteria as:

-Security

-Hardware and software performances

-Resistance to power analysis and other implementation attacks

-Suitability in restricted space environments.

Other point of view is finding and using a methodology for evaluation the computational cost and the complexity of different block ciphers in order to be independent from the platform [6]. That methodology is bridging the gap between the algorithms implementation and mathematical studies.

The main idea was to consider only the amount of the required operations, reducing all the transformations to bitwise-AND and bitwise-OR and shifts.

Software implementation of cryptographic algorithms using the same processor was another kind of analysis and another type of comparison.

The ISO Standard Block Ciphers were compared taking into account their ASIC Performance [15].

For this comparison the basis idea was to research the efficiency of all the known ISO Standard Algorithms in function of the possible implementation for S-Box.

Some other comparison for Block Ciphers was focused on the Hardware Performance [14]. After a general hardware describing for each of the algorithms compact and high-speed hardware architecture were proposed and evaluated.

All algorithms obtained similar performance in compact implementations.

Also, it was proved that GF(((2)2)2)2 inverter is smaller than GF((24)2) by 26%.

One important similar research was done by a team from India in the frame of a project in the Department of Computer Engineering and Information Technology, College of Engineering from Pune, India.

The obtained results of this research were presented in a paper published in the International Journal of Network Security & Its Applications (IJNSA) last year in July [10].

Various features of files like: data density, data types, key size and data size have been analyzed using different symmetric key algorithms. The obtained results concluded that the date size and encryption time is proportional to each other.

In the same time encryption depends only upon the dimension of the file, not upon the data type or density.

At first there will be some general information about these algorithms.

TABLE I. CRYPTOGRAPHYC ALGORITHMS INFORMATION

Algorithm Name	***Structure***	***Key Size (in bits)***	***Rounds***	***Cipher Type***
AES	Substitution – permutation network	128 192 256	10 12 14	Block
CAMELLIA	Feistel Cipher Structure	128 192 256	18 24	Block
SEED	Feistel Cipher Structure	128	16	Block

Following research based on comparing the times required by the three algorithms in different situations depending on the size of the file entry.

To this end have been used 20 files for each algorithm from size 10 KB up to 50MB. For each situation have been running programs and there were times for the three encryption algorithms and for different keys, respectively 128 and 256 (where it exists, i.e. without SEED).

At the same time, account has been taken and that it complies with an averaged over time, so that for each dimension has been tested by 5 runs and it has been calculated mean time.

In this way they were carried out 900 tests for the key for 128-bit and 600 tests for the 256-bit.

Computing systems used were two laptops Asus, both based on microprocessors Intel, Core i5 and Core i7 processors.

To get started there will be a presentation of the three algorithms.

III. EXPERIMENTAL RESULTS

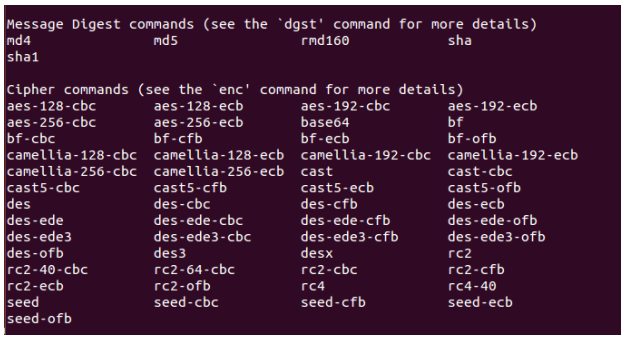

Fig. 1. *An example for command executed in Ubuntu 14.04 LTS*

For the experimental research the Open SSL library included in Ubuntu Linux 14.04 LTS has been used. Linux operating system Ubuntu 14.04 LTS requires minimal maintenance. Computer systems used were two Asus laptops, both based on Intel microprocessors, core i5 and core i7. For testing were used files with the following dimensions: 10KB, 20KB, 30KB, 40KB, 50KB, 100KB, 200KB, 300KB, 400KB, 500KB, 1MB, 2MB, 3MB, 4MB, 5MB, 10MB, 20MB, 30MB, 40MB, 50MB. For each dimension mentioned above were used by 5 files, so the total nuMBer of files that have compared is 100 files. They were recorded the time required to encrypt files and were also calculated and average time on the basis of the foregoing.

For encryption there was chosen a 256-bit key AES and Camellia algorithms (SEED algorithm cannot operate with a 256-bit key) and the version with a 128-bit key for all three algorithms.

128-bit key used is: D3857ABEC68D4

256-bit key used is:

E3C7671A5AD3839AAFBF79DB2596A

The command executed in Ubuntu 14.04 LTS a terminal to encrypt a file using the AES algorithm with a 128-bit key, Open SSL library, followed by the command decryption under the same conditions.

The options available in the library Open SSL for encryption algorithms and hash functions

A. Obtained results using a 128-bit key

The effective obtained time can be seen in the next Excel files. The next two tables show the evolution of encryption time for AES and Camellia and SEED in function of the files dimension.

TABLE II. OBTAINED TIME FOR AES

File lengths	AES					
	File 1	File 2	File 3	File 4	File 5	Average
10 KB	0,004	0,004	0,004	0,004	0,005	0,0042
20 KB	0,012	0,004	0,004	0,004	0,004	0,0056
30 KB	0,004	0,005	0,004	0,004	0,004	0,0042
40 KB	0,004	0,004	0,008	0,004	0,004	0,0048
50 KB	0	0,008	0,004	0,004	0,004	0,004
100 KB	0,006	0,005	0,005	0,006	0,006	0,0056
200 KB	0,007	0,007	0,007	0,005	0,006	0,0064
300 KB	0,007	0,007	0,008	0,007	0,008	0,0074
400 KB	0,007	0,006	0,008	0,008	0,008	0,0074
500 KB	0,008	0,009	0,009	0,008	0,008	0,0084
1 MB	0,012	0,012	0,011	0,012	0,012	0,0118
2 MB	0,018	0,018	0,018	0,02	0,019	0,0186
3 MB	0,026	0,024	0,025	0,026	0,024	0,025
4 MB	0,03	0,032	0,031	0,032	0,035	0,032
5 MB	0,039	0,039	0,038	0,038	0,039	0,0386
10 MB	0,048	0,028	0,036	0,068	0,052	0,0464
20 MB	0,052	0,056	0,056	0,072	0,052	0,0576
30 MB	0,092	0,104	0,076	0,076	0,076	0,0848
40 MB	0,104	0,108	0,096	0,108	0,116	0,1064
50 MB	0,128	0,132	0,12	0,112	0,128	0,124

TABLE III. OBTAINED TIME FOR CAMELLIA

File lengths	Camellia					
	File 1	File 2	File 3	File 4	File 5	Average
10 KB	0,004	0,004	0,006	0,004	0,004	0,0044
20 KB	0,004	0,004	0,004	0,004	0,004	0,004
30 KB	0,008	0,008	0,004	0,008	0,008	0,0072
40 KB	0,008	0,004	0,004	0,008	0,004	0,0056
50 KB	0,008	0,004	0,004	0,004	0,008	0,0056
100 KB	0,007	0,005	0,007	0,007	0,007	0,0066
200 KB	0,009	0,009	0,008	0,009	0,008	0,0086
300 KB	0,01	0,01	0,011	0,01	0,011	0,0104
400 KB	0,012	0,012	0,012	0,012	0,012	0,012
500 KB	0,015	0,014	0,014	0,014	0,014	0,0142
1 MB	0,023	0,023	0,023	0,023	0,023	0,023
2 MB	0,04	0,041	0,041	0,041	0,041	0,0408
3 MB	0,059	0,059	0,059	0,059	0,058	0,0588
4 MB	0,0076	0,076	0,076	0,076	0,076	0,06232
5 MB	0,094	0,094	0,094	0,094	0,094	0,094
10 MB	0,1	0,112	0,112	0,108	0,092	0,1048
20 MB	0,192	0,176	0,188	0,2	0,188	0,1888
30 MB	0,26	0,276	0,284	0,3	0,272	0,2784
40 MB	0,372	0,368	0,36	0,356	0,372	0,3656
50 MB	0,46	0,44	0,436	0,452	0,436	0,4448

TABLE IV. OBTAINED TIME FOR SEED

File lengths	SEED					
	File 1	File 2	File 3	File 4	File 5	Average
10 KB	0,008	0,004	0,004	0,004	0,004	0,0048
20 KB	0,008	0,008	0,008	0,008	0,008	0,008
30 KB	0,008	0,008	0,004	0,008	0,008	0,0072
40 KB	0,008	0,008	0,008	0,008	0,008	0,008
50 KB	0,008	0,008	0,004	0,008	0,008	0,0072
100 KB	0,008	0,008	0,008	0,008	0,008	0,008
200 KB	0,011	0,01	0,011	0,011	0,011	0,0108
300 KB	0,014	0,014	0,014	0,014	0,013	0,0138
400 KB	0,017	0,016	0,016	0,017	0,017	0,0166
500 KB	0,019	0,02	0,02	0,019	0,02	0,0196
1 MB	0,036	0,035	0,034	0,035	0,035	0,035
2 MB	0,064	0,063	0,063	0,063	0,063	0,0632
3 MB	0,092	0,093	0,093	0,092	0,093	0,0926
4 MB	0,121	0,122	0,121	0,122	0,121	0,1214
5 MB	0,151	0,151	0,15	0,151	0,151	0,1508
10 MB	0,2	0,184	0,184	0,188	0,196	0,1904
20 MB	0,356	0,384	0,408	0,384	0,388	0,384
30 MB	0,568	0,544	0,572	0,556	0,592	0,5664
40 MB	0,78	0,748	0,768	0,788	0,776	0,772
50 MB	0,944	0,968	0,94	0,944	0,952	0,9496

The next two diagrams show the evolution of encryption time for Camellia and SEED in correlation to the files dimension.

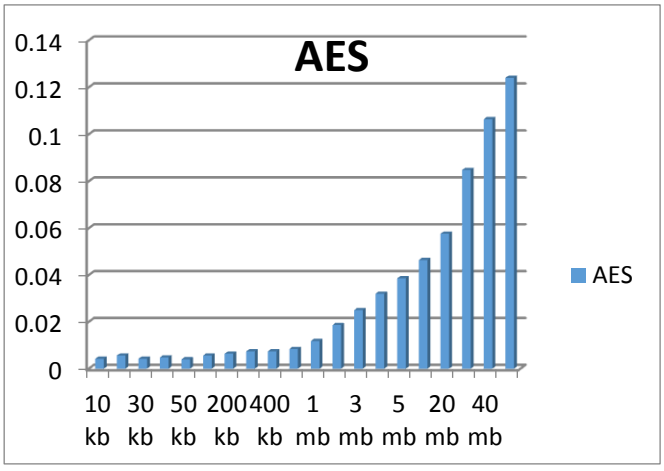

Fig. 2. *AES graph for the above table*

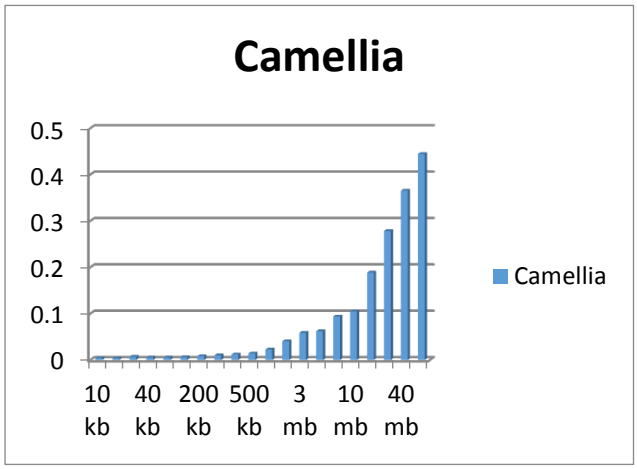

Fig. 3. *Camellia graph for the above table*

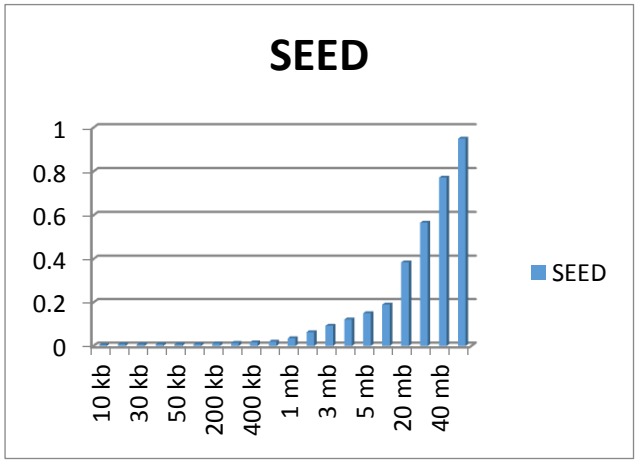

Fig. 4. *SEED graph for the above table*

The following graph provides a better comparison of the three times required encryption algorithms whole set of files.

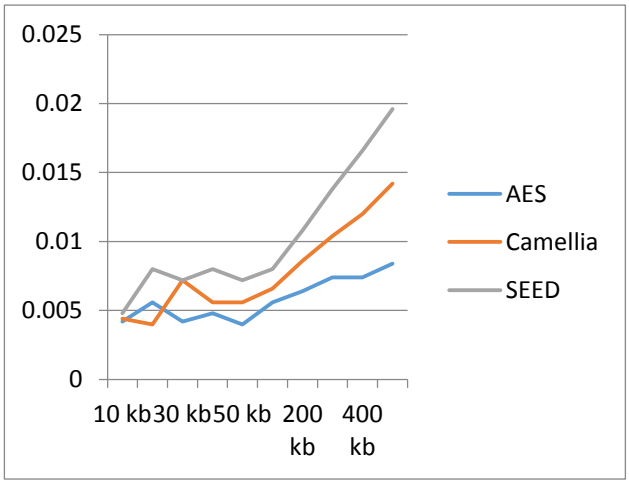

Fig. 5. *Comparison for AES, Camellia and SEED for an 128-bit key*

The following chart shows the same comparison, but considering only the top 10 file size, i.e. files between 10KB and 500KB.

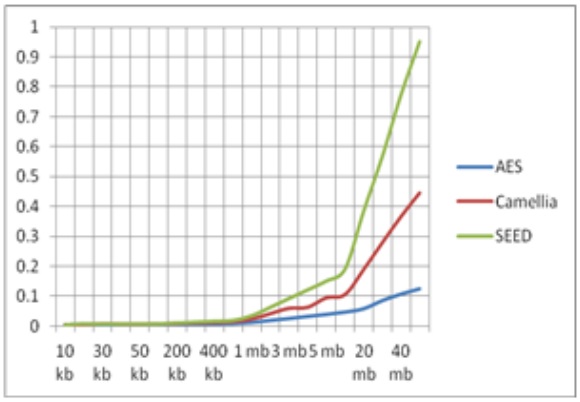

Fig. 6. *Comparison graph for AES, Camellia and SEED for 10 file size*

The graph below shows a comparison of time required encryption of files, taking into account the average of the three algorithms.

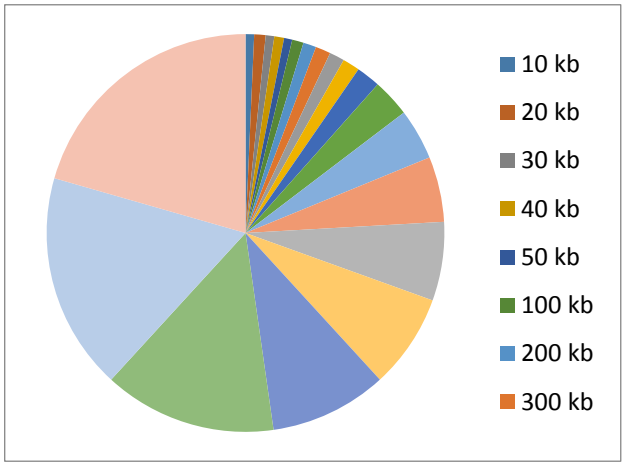

Fig. 7. *Comparison graph for AES, Camellia and SEED using an 128-bit key*

The last three charts show time required for file encryption 100KB, 10MB, 50MB, the four-century considering all the data.

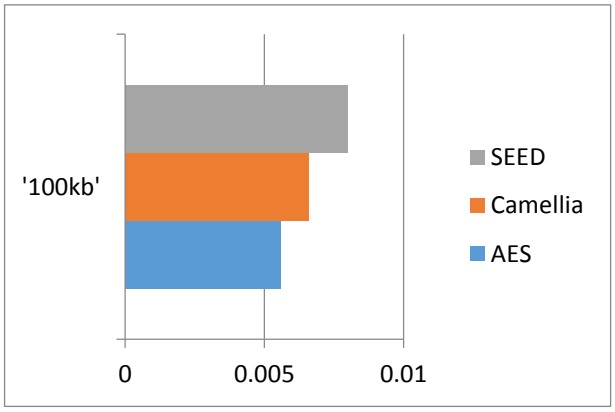

Fig. 8. *Comparison of AES, Camellia and SEED for 100 KB file*

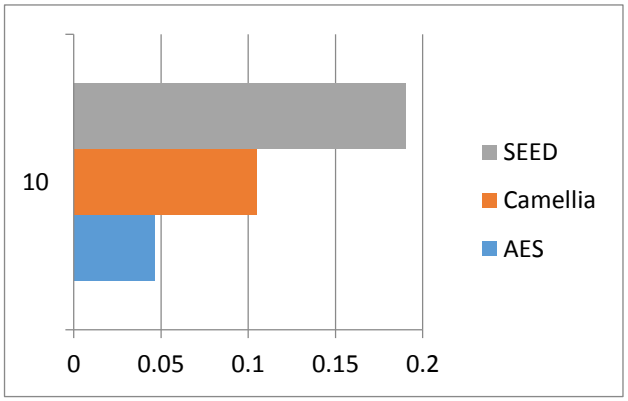

Fig. 9. *Comparison of AES, Camellia and SEED for 10 MB file*

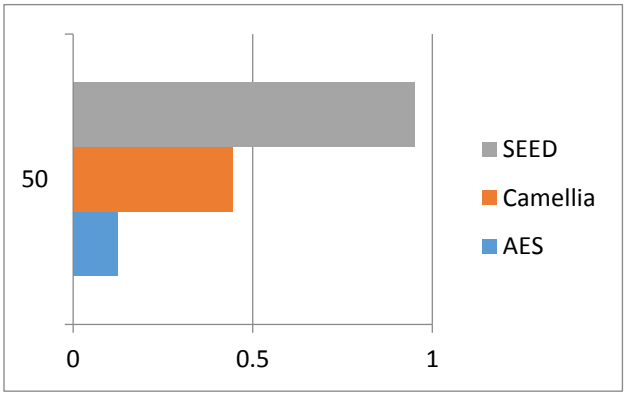

Fig. 10. *Comparison of AES, Camellia and SEED for 50 MB file*

The conclusion is that for encrypting with 128-bit key the AES algorithms is the fastest one, next to it coming Camellia and the slowest from the three is SEED. For small files, between 10 KB and 20 KB Camellia Algorithm was faster than AES. Finally, AES became the most performant one and the difference between it and the others two increase almost exponential for large files (more than 30 KB). The average time for file encryption increases approximately linearly with size.

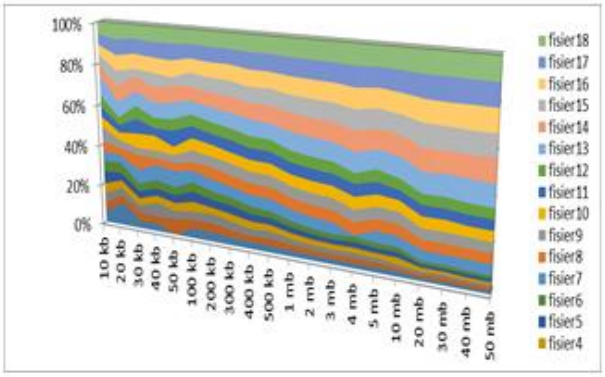

Fig. 11. *Comparison of AES, Camellia and SEED for 50 KB file*

B. Obtained results using a 256-bit key

The times actually obtained were switched auxiliary Excel file accompanying documentation. To present the results still offering some graphic performed on the data from the file.

The next graph shows a comparison between the encryption times for the input files, taking into account an average of the two algorithms.

TABLE V. OBTAINED TIME FOR AES FOR A 256-BIT KEY

File lengths	AES					
	File 1	File 2	File 3	File 4	File 5	Average
10 KB	0,005	0,005	0,005	0,005	0,005	0,005
20 KB	0,005	0,005	0,005	0,005	0,005	0,005
30 KB	0,005	0,005	0,005	0,005	0,005	0,005
40 KB	0,005	0,005	0,005	0,005	0,005	0,005
50 KB	0,005	0,005	0,005	0,005	0,006	0,0052
100 KB	0,006	0,006	0,006	0,006	0,006	0,006
200 KB	0,007	0,006	0,006	0,007	0,006	0,0064
300 KB	0,008	0,007	0,008	0,007	0,007	0,0074
400 KB	0,008	0,007	0,008	0,008	0,007	0,0076
500 KB	0,008	0,007	0,008	0,008	0,009	0,008
1 MB	0,012	0,011	0,012	0,011	0,011	0,0114
2 MB	0,019	0,017	0,017	0,016	0,015	0,0168
3 MB	0,024	0,022	0,021	0,022	0,024	0,0226
4 MB	0,03	0,027	0,028	0,026	0,027	0,0276
5 MB	0,032	0,033	0,032	0,032	0,031	0,032
10 MB	0,056	0,057	0,056	0,057	0,057	0,0566
20 MB	0,11	0,109	0,109	0,111	0,112	0,1102
30 MB	0,162	0,162	0,161	0,16	0,16	0,161
40 MB	0,215	0,212	0,214	0,212	0,215	0,2136
50 MB	0,266	0,265	0,268	0,263	0,263	0,2658

TABLE VI. OBTAINED TIME FOR CAMELLIA FOR A 256-BIT KEY

File lengths	Camellia					
	File 1	File 2	File 3	File 4	File 5	Average
10 KB	0,005	0,005	0,005	0,005	0,005	0,005
20 KB	0,006	0,005	0,005	0,005	0,005	0,0052
30 KB	0,005	0,005	0,005	0,006	0,005	0,0052
40 KB	0,006	0,005	0,005	0,006	0,006	0,0056
50 KB	0,005	0,006	0,006	0,005	0,006	0,0056
100 KB	0,007	0,007	0,007	0,007	0,006	0,0068
200 KB	0,008	0,008	0,008	0,008	0,008	0,008
300 KB	0,009	0,009	0,009	0,009	0,009	0,009
400 KB	0,01	0,011	0,011	0,01	0,01	0,0104
500 KB	0,013	0,012	0,012	0,012	0,012	0,0122
1 MB	0,019	0,019	0,019	0,019	0,019	0,019
2 MB	0,034	0,034	0,034	0,033	0,034	0,0338
3 MB	0,047	0,047	0,047	0,047	0,047	0,047
4 MB	0,06	0,06	0,06	0,06	0,061	0,0602
5 MB	0,074	0,074	0,074	0,075	0,074	0,0742
10 MB	0,151	0,151	0,151	0,149	0,151	0,1506
20 MB	0,295	0,294	0,295	0,294	0,295	0,294
30 MB	0,438	0,435	0,438	0,436	0,438	0,437
40 MB	0,584	0,58	0,587	0,583	0,58	0,5828
50 MB	0,732	0,722	0,724	0,725	0,726	0,7258

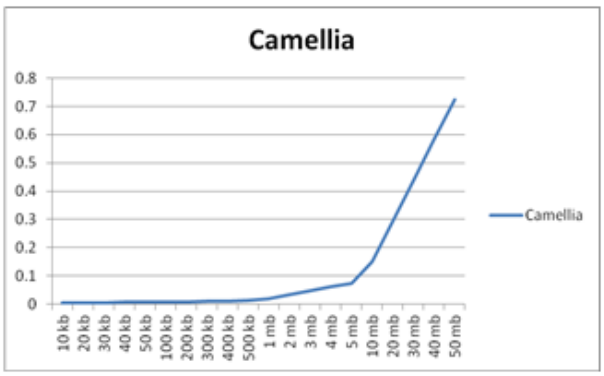

Fig. 12. *Camellia graph for the above table*

The next chart represents a comparison between encryption times for both algorithms for the entire set of input files.

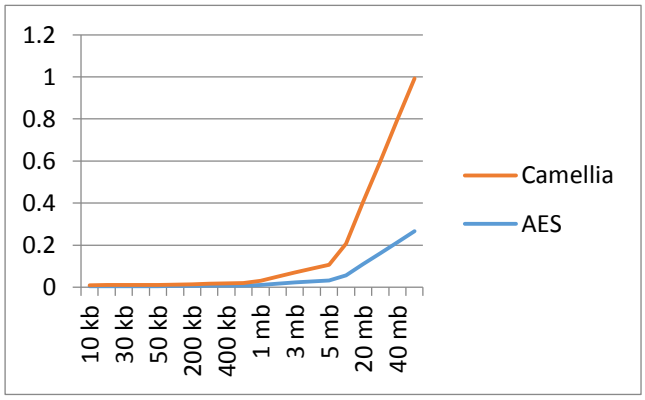

Fig. 13. *Comparison of AES and Camellia for a 256-bit key*

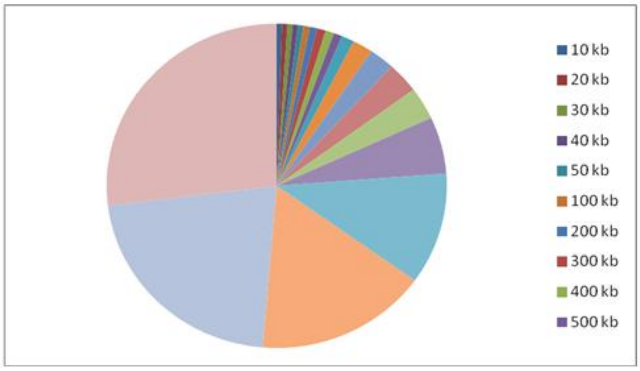

Fig. 14. *Comparisonbetween time needed to encrypt files, taking into account an average of all of the 3 algorithms*

The first three graphs represent the evolution times for each of the two encryption algorithms with increasing file size.

In conclusion, using 256-bit keys, AES algorithm is faster for all sizes of files considered. Also for files over 5MB difference between the two algorithms increases almost exponentially in terms of time required for encryption.

C. Obtained results comparing tests with the 128-bit key and the 256-bit key

Using 256-bit keys is recommended due to increased security that ensures encryption, compared with a 128-bit key. As regards encryption times, the differences are not very big results. A surprising fact occurs in large files, for which time key encryption and 128-bit appeared to be larger than encryption times 256-bit keys. In practical implementations, there are differences and key generation algorithm, a key generation algorithm and 128-bit algorithm is faster than generating a 256-bit keys.

TABLE VII. HARDWARE INFORMATION

		SEED	AES
Algorithm specification	Block length Key length # of rounds	128-bit 128-bit 16	128-bit 128, 129, 256-bit 10(for 128-bit mode)
	Area	3,893 gates 8k-bit ROM	9,917 gates 32 k-bit ROM
Implemental Hardware Complexity	Critical path delay (clock frequency) # of clocks Total time for a 128-bit Encryption/decryption	10,9 ns (90 MHz) 48 533 ns 240 bps	9,9 ns (100 MHz) 11 110 ns 1,16 Gbps

IV. CONCLUSIONS

In this paper there has been presented a comparative analysis of the three encryption algorithms AES, Camellia and SEED.

In our own day has a particular importance comparing main three encryption ciphers for knowing exactly which algorithm is more efficient depending on the size of the file encryption.

AES (Rijndael) with three possible key lengths (128 bit, 192 bit and 256 bit) provides a very high security and very fast software and hardware implementations.

A comparison was done between the three specified ISO/IEC 18033-3-2010, 128-bit block ciphers: AES, Camellia and SEED. After research some practical aspects have been measured.

REFERENCES

[1] S. M. Metev and V. P. Veiko, Laser Assisted Microtechnoogy, 2nd ed., R. M. Osgood, Jr., Ed. Berlin, Germany: Springer-Verlag, 1998.

[2] Aoki K., et al .,"Specification of Camellia – a 128-bit Block Cipher Version 2.0", Nippon Telegraph and Telephone Corporation and Mitsubishi Electric Corporation,2000.

[3] Aoki K., T. Ichikawa, M. Kanda, M. Matsui, S. Moriai, J. Nakajima, and T. Tokita, "Specification of Camellia: A 128-bit Block Cipher", Nippon Telegraph and Telephone Corporation and Mitsubishi Electric Corporation, 2000.

[4] Buchholz J., MATLAB Implementation of the Advanced Encryption Standard, Germany, 2001.

[5] "Camellia Cipher Suites for TLS", June 2010, http://www.rfc-editor.org/info/rfc5932

[6] Daemen J., Rijmen V., "The Design of Rijndael: AES - The Advanced Encryption Standard", Springer-Verlag, 2002.

[7] Granelli F. and Boato G., A novel Methodology for analysis of the computational complexity of Block Ciphers: Rijndael, Camellia And Shacal-2 Compared, University of Trento, Italy, January 2004.

[8] ISO/IEC 18033-3 "Information technology- Security techniques- Encryption algorithms- Part 3: Block ciphers", Jul. 2005.

[9] Jamil T., "The Rijndael Algorithm – A brief introduction to the new encryption standard," IEEE Potentials, Vol. 23, No. 2, April/May 2004, pp. 36 - 38.

[10] KISA, "SEED Algorithm Specification".

[11] Masram R., Shahare V., Abraham J., Moona R., Analysis and comparison of symmetric key cryptographic algorithms based on various file features, International Journal of Network Security &Its Applications (IJNSA), Vol.6, No 4, July 2014.

[12] Mathur M. , Kesarwani A., Comparison between DES, 3DES, RC2, RC6, Blowfish and AES, Proceedings of National Conference on New Horizons in IT-NCNHIT 2013, ISBN 978-93-82338-79-6.

[13] "NESSIE project announces final selection of crypto algorithms", February 27, 2003.

[14] "NESSIE Security Report " vers 2.0.,February 2003.

[15] Satoh A., Sumio Morioka, Hardware-Focused Performance Comparison for the Standard Block Ciphers AES, Camellia, and Triple-DES. Conference: Information Security, 6th International Conference, ISC 2003, Bristol, UK, October 1-3, 2003, Proceedings.

[16] Sugawara T.,Homma N.,Aoki T., and Satoh A., ASIC Performance Comparison for the ISO Standard Block Ciphers,Tokyo, Japan, 2007.

[17] http://www.sourcecodeonline.com/sources/c_c__.html

[18] W.Dai "Crypto++5.2.1 Benchmarks".,2009

[19] http://www.sourcecodeonline.com/list?q=seed_cipher_algorithm

[20] http://www.ipa.go.jp/security/rfc/RFC4312EN.html