How to Assess Security and Efficiency of Authenticated Encryption Algorithms

S.E. ULUSOY[1, 2], O. KARA[1] and M.Ö. EFE[2]

[1] TÜBİTAK, Kocaeli/Turkey, erdem.ulusoy@tubitak.gov.tr
[1] TÜBİTAK, Kocaeli/Turkey, orhun.kara@tubitak.gov.tr
[2] Hacettepe University, Ankara/Turkey, onderefe@ieee.org

***Abstract* – Authenticated encryption is a special form of cryptographic system providing two main services at the same time with a single key: confidentiality and authentication. In 2013, ICRC called authenticated encryption candidates to the CAESAR competition to define a widespread adaptable authenticated encryption algorithm having advantages over AES-GCM. In this study, to analyze competing algorithms, we constitute an extensive metric set by reviewing previous studies and candidate cipher reports. We constitute a metric set composed of all structural metrics mentioned in previous studies. Then, we develop a grading policy for each metric and evaluate ciphers' performance and security. Improvable parts of cipher structures are deduced and listed. Finally, possible future work suggestions are listed to extend metric list and to design better cipher structures.**

***Keywords* – Authenticated Encryption, Security Evaluation, Performance Evaluation, CAESAR Competition, Symmetric Cipher**

I. Introduction

In a confidential communication, secrecy and authenticity of the message must be ensured. In conventional method, confidentiality is provided by encryption algorithms and authentication is supplied by either digital signatures or Message Authentication Codes (MACs). In this method, two different structures are required for confidentiality and authentication with necessity of two different secret keys. These two separated operations overwhelm the confidential communication system. To increase both hardware and time efficiency of confidential communication systems, authenticated encryption [1] is introduced. In an authenticated encryption structure, a single algorithm is implemented to provide confidentiality, integrity and authenticity of a message by using a single key at a time. Using a single structure for two functionalities makes authenticated encryption compact and efficient. Due to its benefits, it is adopted and used broadly in encryption systems where authentication is also required.

Strength and functionality of an encryption standard must be ensured before becoming wide-spread. Assessment of an encryption algorithm isn't an easy work. Difficulty of strength and functionality assessment and development of an encryption algorithm has been observed, hence before spreading an algorithm world-wide, competitions are run around the world to develop and analyze the candidate algorithms. Some examples of these competitions are AES competition of NIST to standardize a strong encryption algorithm, SHA-3 competition to standardize a hashing algorithm.

In 2013, the CAESAR competition [2] (Competition for Authenticated Encryption: Security, Applicability and Robustness) was organized. CEASAR competition is the first competition to evaluate AEAD algorithms and at the moment, how to evaluate AEAD isn't totally clear. In [3], aim of the competition is defined as to determine a portfolio of widespread adaptable authenticated encryption algorithms having advantages over AES-GCM [4] (Advanced Encryption Standard – Galois Counter Mode) by International Cryptographic Research Community (ICRC). In the competition, 3 different use cases are defined: 1. High Performance Applications, 2. Lightweight Applications, 3. High Security Applications. 57 algorithms have applied to competition and at the moment 7 of them are running in the final round and waiting for the announcement of final portfolio.

During these competitions, the candidates are reviewed, investigated and analyzed for comparison with each other and figuring out any possible weaknesses in design. For example, in [5, 6], lightweight ciphers are reviewed and compared. In [7], Abed et.al reviewed and classified Round 1 CAESAR candidates according to their performance, security and implementations.

In this study, to propose a method to evaluate AEAD algorithms, we gather metrics from different studies to constitute the most extensive metric set so far. We keep out of scope only robustness to side channels attacks and strength of PRP (pseudorandom permutation) and PRF (pseudorandom function) since they require special analyses that needs to be exclusively studied. Also to the best of our knowledge, our study is the first study, giving weights to metrics based on use case and introducing a grading policy for the metrics. Competitors are analyzed and graded according to the determined metrics. After evaluating the ciphers, we list the metrics where ciphers lose points and discuss how they can avoid losing those points. Finally, the study is summarized and possible future work suggestions are listed.

The rest of this paper is organized as follows: In Section II, we introduce final round candidates of the CAESAR competition. In Section III, we explain the metrics, their rationale and our grading policy. In Section IV, we score the algorithms and explain the reasoning behind. In section V, we compare and analyze where the algorithms lose points. In Section VI, we give general recommendations to increase performance and security of authenticated encryption algorithms based on how they lose points. In Section VII, we conclude the study and mention the possible future research issues.

II. The CAESAR Finalists

There are 7 algorithms competing in the final round of the CAESAR competition. The finalists are distributed widely based on their structure (3 block ciphers, 3 state ciphers and 1 LFSR (linear feedback shift register).

All three block cipher algorithms use AES algorithm as block function. First one is COLM [8] which is an encrypt-mix-decrypt (EMD) construction. Second is Deoxys [9], a tweakable block cipher based on offset codebook (OCB) mode of AES. Final block cipher is OCB [10].

The first state cipher is AEGIS [11] using AES rounds as the state update function. Another state cipher is MORUS [12] using basic bit-rotations, AND and XOR operations in the state update function. The last one is ASCON [13] which uses a sponge construction, a special form of state cipher. In a sponge construction, a state value is hold but during encryption and decryption a single branch of the state is used to encryption and other branches are only transferred to the next state function.

The last finalist is ACORN [14], a stream cipher built by cascading 6 different LFSRs. As a stream cipher ACORN is the lightest cipher among the other finalists.

III. Assessment Metrics

Determining the metrics is the most critical and challenging part of this study because any structural metric mustn't be missed and metric points must be determined carefully for a fair comparison method. While determining the metrics, we first search similar studies in the literature. In [6], Abed et.al. classified the CAESAR competitors based on their construction methods, operation modes, masking methods and functional characteristics. They also reviewed attacks performed on candidates. They only classified the ciphers and created tables showing if the algorithms have the listed functional characteristics. We start to construct our metric set by using the metrics in their study. Then, we review design rationale and features of the canditate ciphers and add appropriate properties as metrics to our metric set. After finishing our metric list, we determine the metric strengths for three use cases: High Performance, Lightweight and High Security. Generally, the metric strengths are determined on 3 possible values: N.A., out of 5 and out of 10. We stay sticked to these three values as much as possible not to lose fairness of metric points. Metric strengths are shown in Table 1.

The definitions and grading of the metrics are as follows:

The first 5 metrics are security related metrics since security is the primary concern in an encryption algorithm. Hence, for high performance and lightweight use cases, ciphers are graded out of 5 instead of calling these metrics N.A.

1. Replaceable PRP (Pseudorandom Permutation) and PRF (Pseudorandom Function): Cryptanalysis techniques improve day by day so in the future current PRP may not be secure anymore and need to be replaced.

a. Not replaceable. For all cases: **Score is 0.**

b. Replaceable. For High Security: **Score is 10.** For other use cases: **Score is 5.**

Table 1: Metric Strengths.

Metric	High Performance Use Case	Lightweight Use Case	High Security Use Case
1. Replaceable PRP and PRF	Out of 5	Out of 5	Out of 10
2. Natural Resistance to CCA&CPA	Out of 5	Out of 5	Out of 10
3. Domain separation between AD and PT	Out of 5	Out of 5	Out of 10
4. Strength of "Nonce/Tweak/IV"	Out of 5	Out of 5	Out of 10
5. Difference between two ciphertexts	Out of 4	Out of 4	Out of 8
6. Necessity of decrypting message to check authentication	Out of 10	Out of 10	Out of 10
7. Effect of fixed use or reuse of AD	Out of 10	Out of 10	Out of 10
8. Incremental AD Process and Authenticated Encryption	Out of 5	Out of 5	N.A.
9. Cipher Overhead	Out of 10	Out of 10	N.A.
10. Being Parallelizable	Out of 10	Out of 5	N.A.
11. Being Online	Out of 5	Out of 10	N.A.
12. Being two-pass or single-pass	Out of 5	Out of 5	N.A.
13. Being inverse-free	N.A.	Out of 10	N.A.
Total	Out of 79	Out of 89	Out of 68

2. Natural Resistance to CCA (Chosen Ciphertext Attack) &CPA (Chosen Plaintext Attack): In cryptanalysis techniques, it is assumed that the attacker owns the oracle and is able to use it with the embedded secret key. Two common attacks to recover secret key based on this assumption are CCA and CPA. If the cipher doesn't work with a random ciphertext or creates low correlated ciphertexts when plaintexts are given we can say, the algorithm has natural resistance to CCA and CPA, respectively.

a. No resistance to neither CCA nor CPA. For all cases: **Score is 0.**

b. Resistance to either CCA or CPA. For High Security: **Score is 5.** For other use cases: **Score is 3.**

c. Resistance to both CCA & CPA. For High Security: **Score is 10.** For other use cases: **Score is 5.**

3. Domain separation between AD and PT (plaintext): If an attacker obtain the cipher, they may manipulate the oracle by changing roles of AD Blocks and PT blocks. The algorithm must hinder any possible attack done this way. This is generally achieved by domain separation between AD and PT.

a. No domain separation between AD process and

Encryption. For all cases: **Score is 0.**

b. Domain separation between AD process and Encryption. For High Security: **Score is 10.** For other use cases: **Score is 5.**

4. Strength of "Nonce/Tweak/IV". A weak "Nonce/Tweak/IV" may weaken a strong cipher significantly so it is an important issue for security. In this metric three conditions will be analyzed: (i) unpredictability, (ii) ease of production and (iii) effect on the authentication and security levels when changed.

For satisfying conditions (i) and (ii), a cipher gets 2 points from each condition in the high security use case, and 1 point from each condition in other use cases.

Condition (iii) is evaluated in 3 levels: less than or equal to birthday attack limit, more than birthday attack limit and the highest level security. In the high security case, a cipher gets two points for each level and a single point in the other use cases.

5. Difference between two ciphertexts: In this metric, there are 4 evaluation levels:

a. An existing relation (such as: CT1 XOR CT2 = PT1 XOR PT2) between ciphertext and plaintext couples.

b. Plaintexts having the same parts are encrypted to ciphertexts having same parts

c. Plaintexts having the same beginnings are encrypted to ciphertexts having the same beginnings

d. Ciphertexts are totally uncorrelated in any case

For the high security use case, a cipher gets two points for each level, and a single point for the other use cases.

6. Necessity of decrypting the message before checking authentication: If a message with an invalid tag is decrypted, this may cause a security risk and waste resources.

a. Ciphertext must be decrypted totally with leakage risk of newly generated plaintext: **Score is 0.**

b. Ciphertext must be decrypted partially with a leakage risk of newly generated partial plaintext: **Score is 5.**

c. Authentication can be done without decrypting the ciphertext: **Score is 10.**

7. Effect of fixed or reused AD. Fixed or reused AD mustn't alter authentication or security levels of a cipher.

a. Fixed or reused AD decreases both authentication and security levels: **Score is 0.**

b. Fixed or reused AD decreases authentication level but doesn't affect security level: **Score is 5.**

c. Fixed or reused AD doesn't affect neither authentication nor security levels: **Score is 10.**

8. Incremental associated data process and authenticated encryption (incremental AEAD): In authenticated encryption with associated data, two subsequent messages (M, M') may differ by just a fraction. In that case, if the ciphertext and tag pair (C, T) is given for M, then (C', T') for M' can be computed in a more efficient way than encrypting M' from scratch. This reduces computation cost and increases the performance of the system. (For High Security: N.A.)

a. If incremental AEAD isn't possible: **Score is 0.**

b. If incremental AEAD is possible: **Score is 5.**

9. Cipher overhead: Overhead means extra work so it is undesirable for a better performance and source usage. This metric is analyzed for two cases: overhead per block and overall overhead. (This metric is N.A. for high security use case)

- Per data block: (Out of 6)

a. More than twice per block **Score is 0.**

b. Twice per block. For High Performance: **Score is 3.** For Lightweight: **Score is 2.**

c. Once per block. For High Performance: **Score is 6.** For Lightweight: **Score is 4.**

d. Better than once per block. For High Performance: **Score is 6.** For Lightweight: **Score is 6.**

- Overall: (Out of 4)

a. 1 point is deducted per overhead, up to 4 points.

10. Being parallelizable: Ciphers' structures vary from sequential to fully parallelizable (where all data blocks can be computed together in parallel). If there is no maximum point limit, this metric would dominate other metrics and result would be highly dependent on this metric. To prevent dominance of a single metric and keep balance between metrics, the maximum point is limited to 10 (the highest possible point for other metrics) for High Performance Use Case and 5 for Lightweight Use Case. Also to span a wider range, we choose logarithmic scale instead of linear and double the base after 16 parallel computations for High Performance Use Case and after 8 parallel computations for Lightweight Use Case. Lightweight Use Case has a tighter grading policy because limited resources make parallelization harder: (For High Security: N.A.)

a. For High Performance:

```
if (n >= 65536)
    score = 10
else if  (n <= 16)
    score = $\log_2(n)$
else
    score = $4+\log_4(n/16)$
```

b. For Lightweight:

```
if (n >= 128)
    score = 5
else if  (n <= 8)
    score = $\log_2(n)$
else
    score = $3+\log_4(n/8)$
```

11. Being online: An online cipher can process the data without waiting to receive whole data. It is obvious that an online cipher has a better performance than an offline cipher. Also for Lightweight Use Case, an online cipher reduces the amount of memory to buffer data while receiving. (For High Security: N.A.)

a. If the cipher is not online: **Score is 0.**

b. If the cipher is online: For High Performance: **Score is 5.** For Lightweight: **Score is 10.**

12. Being two-pass or single-pass. Two-pass means extra work on ciphertext so an efficient algorithm is expected to be single-pass. (For High Security: N.A.)

a. If the algorithm is two-pass: **Score is 0.**

b. If the algorithm is single-pass: **Score is 5.**

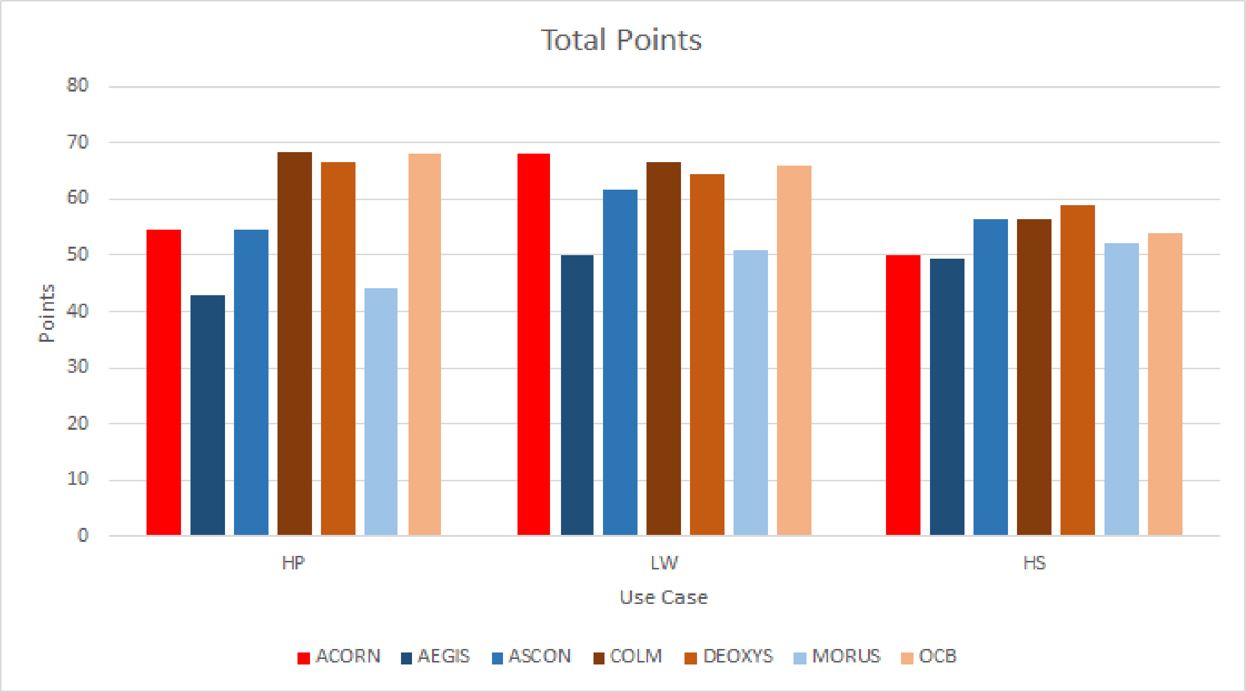

Figure 1. Total Points of Ciphers for 3 Use Cases

13. Being inverse-free. Being inverse-free doesn't have any effect on neither performance nor security but for Lightweight Use Case, implementing encryption and decryption together would save significant amount of resources.
   a. If the cipher isn't inverse-free: **Score is 0.**
   b. If the cipher is inverse-free: **Score is 10.**

IV. Grading Algorithms

Our grading policy is covering the most of the cases in the practice. On the other hand, some applications doesn't hit the predetermined points. In this section, we explain how miss situations are graded and explained the reasoning.

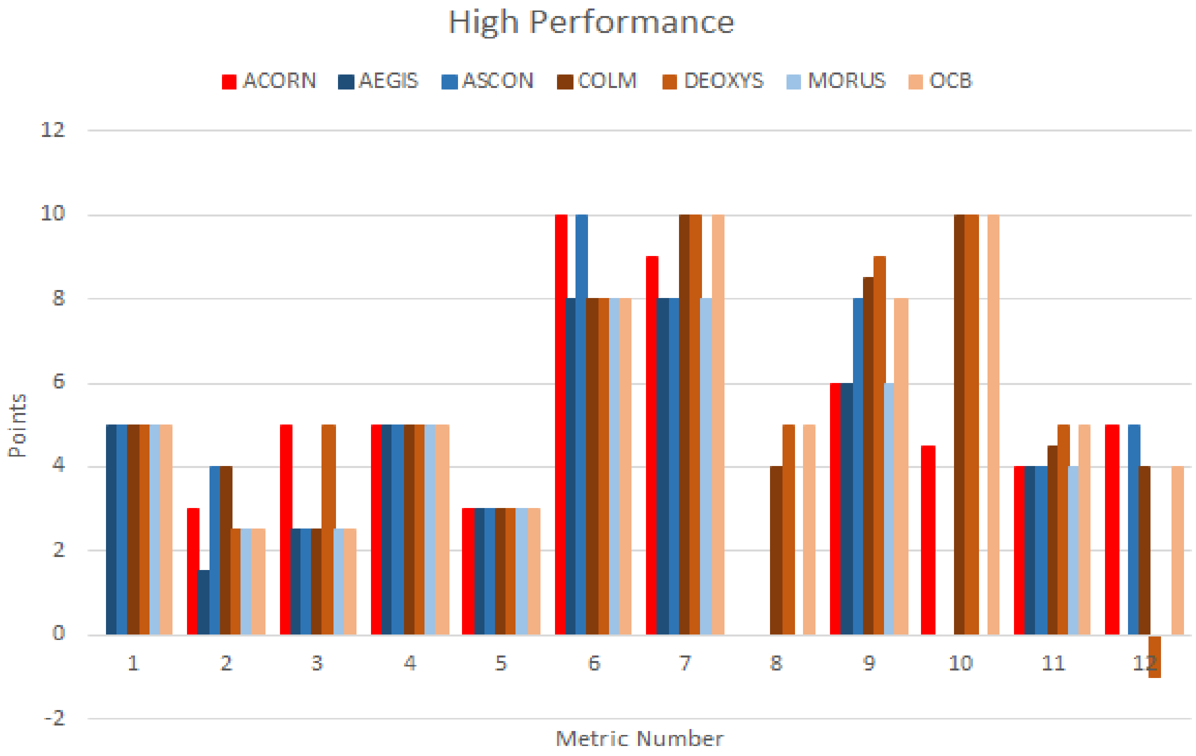

Figure 2. Points per Metric for High Performance Use Case

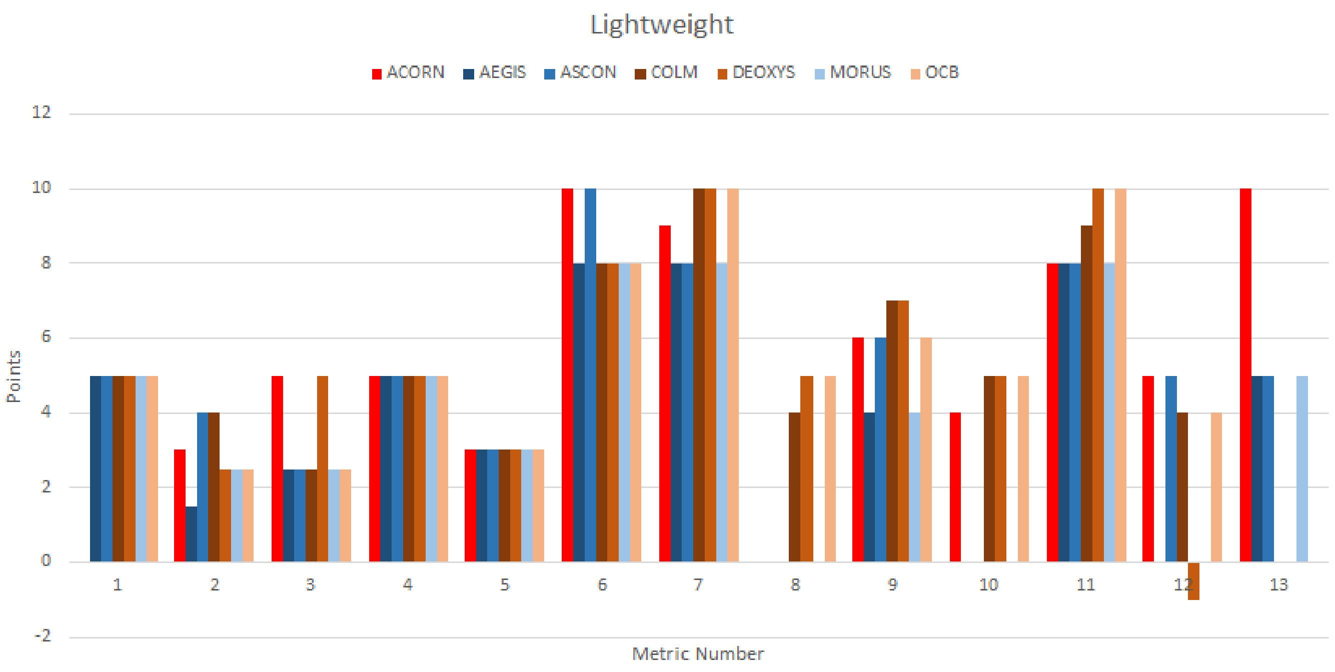

Figure 3. Points per Metric for Lightweight Use Case

Partial point explanations:

1. Metric 2: If the ciphertext is decrypted, then a state for tag is computed and then the plaintext isn't needed to check authentication, partial point is given for CCA. If the cipher has a nonce (npub) which is public and controlled by cipher, partial point is given for CPA.
2. Metric 3: If there isn't a concrete separation between AD and PT blocks but AD and PT blocks can't be used instead of each other directly or using them doesn't leak information to analyze easily, it is considered as a partial separation and a partial grade is given.
3. Evaluation of Metric 6 is combined with evaluation of Metric 12 as follows: if the ciphertext must be decrypted and after a state is calculated to check tag, there is no more need of the plaintext, it is assumed that the cipher doesn't need to decrypt the ciphertext to check authentication but is two-pass.

V. Comparison and Analyses of the Results

To ease the comparison, we divide ciphers into three groups. The first group is LFSRs in which ACORN is the single cipher. The second group is the state based ciphers composed of AEGIS, ASCON and MORUS. The last group is the block ciphers composed of COLM, DEOXYS and OCB.

For High Performance Use Case, from Figure 1, it can be said that block based ciphers give the best results. From Figure 2, the main reasons for this achievement can be listed as Metrics 7, 8 and 10. The Metric 7 is the effect of fixed or reused AD, LFSR and state-based ciphers lose points from this metric because the partial (for LFSR) or first block (for state based) of PT is XORed with same padding. The reason behind why block based ciphers gain points from Metrics 8 and 10 is quite similar to each other. Since they work block-based, parallel computations are possible and if there is a change only in some blocks, it is enough to compute cipher value of only the changed part and its effect on the result.

For Lightweight Use Case, from Figure 1, it can be said that LFSR cipher, ACORN, overwhelms other candidates and places in the first position. As seen from Figure 3, the main reason behind this is that the LFSR cipher, ACORN, works as an inverse-free cipher which is a highly desired property for lightweight applications.

For High Security Use Case, all ciphers have similar grades, as seen from Figure 1. It is because block based ciphers lose their advantage coming from possible parallel computations and LFSR cipher loses its advantage from being inverse free. As seen from Figure 4, ciphers get more or less the same points from other metrics.

Another point to mention is that if the block algorithms of block based ciphers, or state functions of state based ciphers become obsolete due to security related or other reason, they can be replaced without disturbing other parts of the design but if a security related or another problem occurs in the LFSR, the algorithm must be redesigned completely where LFSR cipher loses points.

VI. General Recommendations For Algorithms

In our grading system, LFSR cipher lose the most important points from not having a replaceable PRP and PRF. At the moment it seems hard to overcome this problem. Another point that can be improved for LFSR ciphers is number of possible parallel computations. Final improvable point is during initialization and other intermediate operations; they have excessive computation overhead. For efficiency in small data, their overhead must be reduced.

In the state based ciphers, there is no domain separation between AD and PT, but this doesn't seem to risk the security

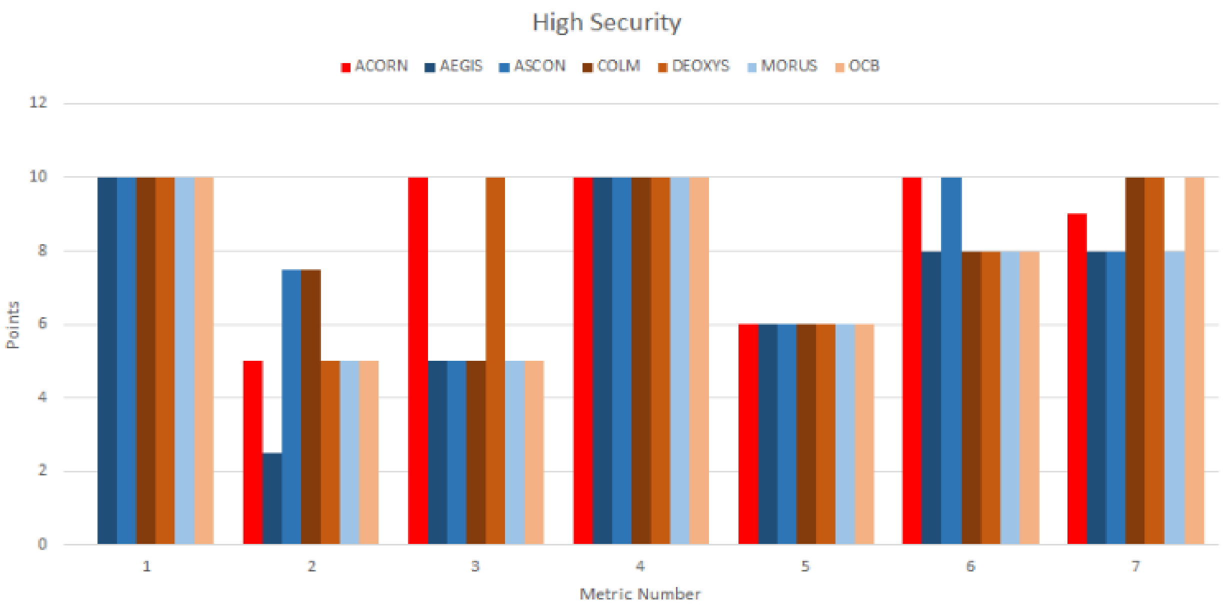

Figure 4. Points per Metric for High Security Use Case

because the state is used to determine padding bits and in the calculation of the next state, the data block being either AD or PT becomes a trivial issue. Another topic on the state based ciphers is that the ciphertext could be used as a state block, as in ASCON – the sponge construction, to check the tag to avoid decryption of the plaintext. The final issue is since they are not parallelizable and have a lot of overhead during initialization, they lose points for performance evaluation.

Block ciphers are being used for a long time, and its effects are seen as high success in the results. A small suggestion for block cipher developers is to build the cipher with more obsolete domain separation between AD and PT because it seems to make block ciphers prone to forgery attacks.

VII. Conclusion and Future Work

In this study, we determine metrics for assessment of AEAD ciphers in a wide concept. Developing better cryptographic constructions always goes on. By this study, we try to show some possible weaknesses in the designs, and ways to cover these weaknesses with related reasoning for developers to help them to construct ciphers with higher security and performance.

In our analysis, we consider only the structures of the algorithms, but PRP and PRF of a cipher and robustness to side channel attcks also play an important role both in security and in performance so their evaluation is as more important as evaluation in this study. We leave analyzing the effect of PRPs and PRFs and to security and performance and robustness to side channel attacks as a future study.

Also based on where and why ciphers lose points, new cryptanalyses can be performed to current algorithms and their security claims can be analyzed.

VIII. References

[1] M. Bellare and C. Namprempre, "Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm," in *International Conference on the Theory and Application of Cryptology and Information Security*, Kyoto, 2000.

[2] D. Bernstein, "https://competitions.cr.yp.to," 07 03 2018. [Online]. Available: https://competitions.cr.yp.to/caesar.html. [Accessed 07 06 2018].

[3] D. Bernstein, "Crypto Competitions: CAESAR call for submissions, final (2014.01.27)," ICRC, 26 04 2014. [Online]. Available: https://competitions.cr.yp.to/caesar-call.html. [Accessed 18 06 2018].

[4] M. Dworkin, "Recommendation for Block Cipher Modes of Operation Galois/Counter Mode (GCM) and GMAC," National Institute of Standards and Technology, 2007.

[5] C. Manifavas, G. Hatzivasilis, K. Fysarakis and Y. Papaefstathiou, "A Survey of Lightweight Stream Ciphers for Embedded Systems," *Security and Communication Networks,* pp. 1226-1246, 2016.

[6] B. J. Mohd, T. Hayajneh and A. V. Vasilakos, "A Survey on Lightweight Block Ciphers for Low-Resource Devices: Comparative Study and Open Issues," *Journal of Network and Computer Applications,* no. 58, pp. 73-93, 2015.

[7] F. Abed, C. Forler and S. Lucks, "General Classification of the Authenticated Encryption Schemes for the CAESAR Competition," *Computer Science Review,* no. 22, pp. 13-26, 2016.

[8] E. Andreeva, A. Bogdanov, N. Datta, A. Luykx, B. Mennink, M. Nandi, E. Tischhauser and K. Yasuda, "Colm v1," 2016.

[9] J. Jean, I. Nikolic, T. Peyrin and Y. Seurin, "Deoxys v1.41," ANSSI, Paris, 2016.

[10] T. Krovetz and P. Rogaway, "OCB (v1.1)," 2016.

[11] H. Wu and B. Preneel, "AEGIS: A Fast Authenticated Encryption Algorithm," Nanyang Technological University, Burnaby, 2014.

[12] H. Wu and T. Huang, "The Authenticated Cipher MORUS (v2)," Nanyang Technological University, Singapur, 2016.

[13] C. Dobraunig, M. Eichlseder, F. Mendel and M. Sclaeffer, "Ascon v1.2 Submission to the CAESAR Competition," Institute for Applied Information Processing and Communications, Graz, 2016.

[14] H. Wu, "ACORN: A Lightweight Authenticated Cipher (v3)," Nanyang Technological University, 2016.