SCIS 2014 The 31th Symposium on Cryptography and Information Security
Kagoshima, Japan, Jan. 21-24, 2014
The Institute of Electronics, Information and Communication Engineers

# Who can survive in CAESAR competition at round-zero?

HakJu Kim * Kwangjo Kim *

**Abstract:** Cryptographic primitives are required to protect an IT (Information Technology) system. They are used to provide CIA (Confidentiality, Integrity, and Availability) and other security attributes to the system. So far, NIST (National Institute of Standard and Technology) has successfully standardized AES (Advanced Encryption Standard) for confidentiality and SHA (Secure Hash Algorithm) for integrity. Authenticated Encryption is a cryptographic primitive or mode that simultaneously provides confidentiality, integrity, and authenticity. CAESAR (Competition for Authenticated Encryption: Security, Applicability, and Robustness), funded by NIST, is a competition for Authenticated Encryption. CAESAR provides a long example list of features that can be used to evaluate the submissions, but there is no public notion that indicates the importance of each feature. This paper analyzes Authenticated Encryption modes submitted to NIST and predict the essential features of the submissions to survive CAESAR competition.

**Keywords:** Authenticated Encryption, CAESAR, NIST

## 1 Introduction

Information security is the art of protecting information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction in order to provide information security attributes like Confidentiality, Integrity, and Availability[1]" which are the most basic security requirements. Confidentiality provides that only authorized user can access the protected information. Integrity is the assurance that the information must be consistent and accurate. Availability is a way of guaranteeing that information is available when needed.

The system security manager implements all information security attributes to assure information security of the system. However, the implementation of each information security attribute will cause the overhead to the system. The overhead can be reduced if multiple information security attributes are provided by one information security approach or algorithm. "Authenticated Encryption is a shared-key based transform whose goal is to provide both privacy and authenticity of the encapsulated data[2]". Authenticated Encryption is a combination of two information security approaches; encryption and authentication. In past, implementers simply glued provably secure encryption and authentication algorithms together, but many of the resulting modes of Authenticated Encryption were insecure and slow[2]. We define such approach as Naïve Composition. The composition methodology of Authenticated Encryption can affect the security and the performance. Therefore, NIST (National Institute of Standards and Technology), which has successfully standardized AES (Advanced Encryption Standard) and SHA (Secure Hash Algorithm), is accepting modes of Authenticated Encryption and posting them on the NIST webpage[3] for public consideration. Modes are schemes that uses existing block ciphers or hash algorithms. Algorithms are entirely different from existing block ciphers or hash algorithms. 14 submissions are listed on the NIST homepage. Furthermore, NIST is funding the competition called CAESAR (Competition for Authenticated Encryption:

* Computer Science Dep't, KAIST, 291 Gwahak-ro, Yuseong-gu, Daejeon, 305-701, Korea. {ndemian, kkj}@kaist.ac.kr

Security, Applicability, and Robustness)[4]. Submission to CAESAR can be either an algorithm or a mode. The due date of the first-round submission for CAESAR is January 15th, 2014. Although CAESAR is not an official standardization competition from NIST, it has drawn much attention globally. Furthermore, CRYPTREC (Cryptography Research and Evaluation Committees)[5], set up by the Japanese Government, evaluates and recommends cryptographic primitives. The outcome of CRYPTREC has contributed in various standards.

The study of Authenticated Encryption without a complete survey or a definite standard is difficult. Modes of Authenticated Encryption submitted to NIST are different in many aspects like structure and performance. To the best of our knowledge, there have been no survey on modes of Authenticated Encryption that are related to CAESAR or the submissions to NIST. We strongly believe that the summary and the analysis of the submissions are essential to recognize strength and weakness of each mode or algorithm. CAESAR provides a comprehensive list of evaluation features, but refuses to state which features are important. The important features will be extracted by observing the existing modes of Authenticated Encryption. The extracted features can be used to predict which submissions can survive CAESAR competition.

## 2 Classification of Authenticated Encryption

The practitioners initially glued the encryption algorithm and the authentication algorithms together to provide encryption and authentication at the same time. We define such composition scheme as Naïve Composition, which resulted in degraded performance and poor security. To provide an efficient and secure scheme, the researchers proposed three composition schemes for Authenticated Encryption; (i) General composition paradigm, (ii) Encrypt with redundancy, and (iii) Encode-then-encipher. Figure 1 shows the classification of Authenticated Encryption. General composition paradigm was the winner among three schemes, because Encrypt with redundancy and Encode-then-encipher are rather insecure and inefficient than General composition paradigm[2][6][7]. Encrypt with redundancy and Encode-then-encipher are crossed out in Figure 1. However, General composition paradigm has one major disadvantage; two separate keys for encryption and authentication are required. The additional key management requires the overhead to the system.

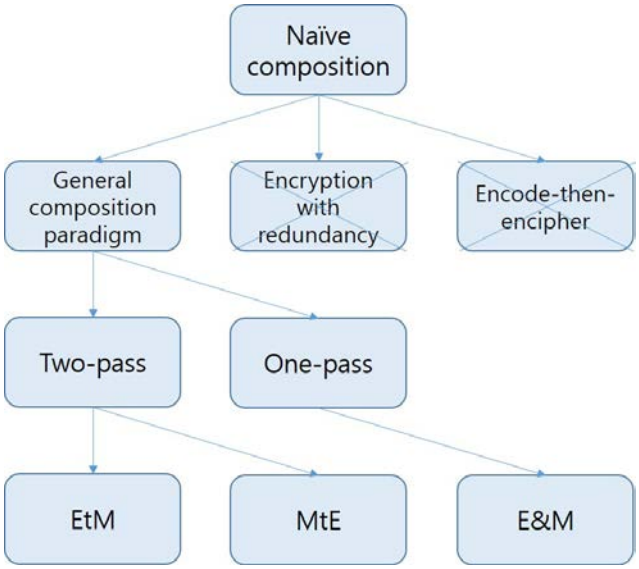

Figure 1: Classification of Authenticated Encryption

General composition paradigm is developed into one-pass and two-pass modes. One-pass mode executes encryption and authentication at the same time; encrypt-and-authenticate (E&M). Two-pass mode executes one algorithm first and then executes another; encrypt-then-authenticate (EtM) or authenticate-then-encrypt (MtE). Most of one-pass and two-pass modes uses one common key for both encryption and authentication. Some modes use one key to generate counter and another key to encrypt and authenticate. Two-pass mode is the variation of General composition paradigm using only one key. Most two-pass modes uses EtM scheme which was proven to be most secure under General composition paradigm. One-pass mode is not directly derived from General composition paradigm. E&M scheme was considered insecure under General composition paradigm, but one-pass mode is the secure version of E&M scheme. Although designing a secure one-pass mode is more difficult than designing a secure two-pass mode, one-pass mode is considerably faster than two-pass mode. Furthermore, the creation of an entirely new Authenticated Encryption algorithm without the existing encryption or authentication algorithm can be feasible.

## 3 Modes of Authenticated Encryption

NIST has accepted 14 modes of Authenticated Encryption and posted the submissions on its webpage for public consideration. The summary and

analysis of Authenticated Encryption will be based on the submissions to NIST, and the key evaluation features will be extracted from the analysis of the submissions. The extracted features should be applicable to find the selection of CAESAR.

We will divide the submissions to NIST into three groups for the sake of simplicity. First group consists of CCM-family, second group consists of IAPM-family, and last group consists of other submissions. The criteria to divide the groups is the relationship between the submissions. The relationship is expressed in Figure 2. The arrow represents which mode has affected another mode.

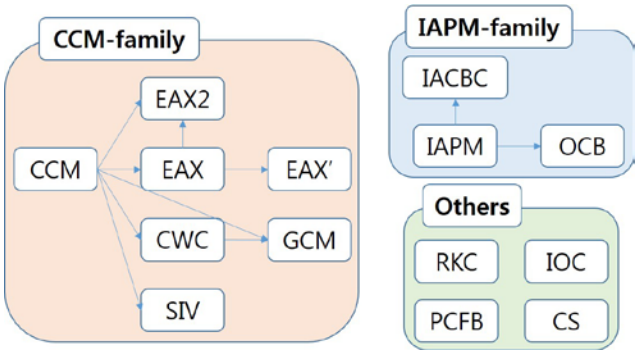


Figure 2: Relationship of the submissions

### 3.1 CCM-family

CCM (Counter with CBC-MAC)[8] is a standard mode of operation which provides both encryption and authentication of given data. Many modes of Authenticated Encryption are derived from CCM with similar structure. Figure 3 shows the structure of CCM. We call the variations of CCM to be CCM-family. The variations in CCM-family are EAX (Encrypt-then-Authenticate-then-Translate)[9], CWC (the Carter-Wegman message authentication scheme with CTR mode of encryption)[10], GCM (Galois / Counter Mode)[11], and SIV (Synthetic IV)[12]. CCM-family has many similarities. First, CCM-family

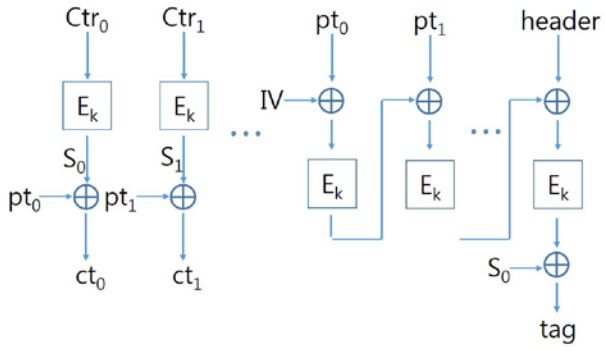


Figure 3: Structure of CCM

uses CTR mode for encryption (*e.g.* AES), so they can decrypt the ciphertext using the encryption circuit. Second, except for SIV, CCM-family uses two-pass EtM mode; SIV uses two-pass MtE mode. Encryption part of CCM-family is fully parallelizable in block-level. Third, CCM-family has some preprocessing capability at encryption part. Fourth, CCM-family has similar memory requirement. CCM-family typically has relatively lower memory requirement, but CWC and GCM may require more memory to increase the performance. Fifth, CCM-family is on-line; the encryption and authentication can begin before whole message is arrived. Sixth, CCM-family is patent-free. Last, CCM-family has provable security, but CCM has lower security level than others; mode developers have criticized the security of CCM in their submissions.

However, the variations of CCM-family have different features. First, CCM-family uses different authentication algorithms to generate MAC (Message Authentication Code). CCM uses CBC-MAC (Cipher Block Chaining MAC), EAX uses OMAC (One-keyed MAC), EAX prime[13], an improved version of EAX uses CMAC (Cipher-based MAC), CWC uses the Carter-Wegman universal hashing, GCM uses the universal hashing under binary Galois field, and SIV uses CMAC to authenticate. Second, CCM-family produces different MAC length (or tag length), and require different length of nonce, initialization vector, and counter. Last, CCM-family has different parallelizability in authentication part. Authentication part of CWC and GCM is parallelizable in bit-level, but authentication part of the others is not parallelizable.

**Note**: EAX2[9] is a general composition paradigm version of EAX. EAX2 uses two symmetric keys for encryption and authentication.

### 3.2 IAPM-family

IAPM (Integrity Aware Parallelizable Mode)[14] is a one-pass E&M mode. IACBC (Integrity Aware CBC)[15] and OCB (Offset Codebook)[16] are derived from IAPM. Figure 4 shows the structure of IAPM. IAPM-family shares similar structure. IAPM and OCB uses only an encryption algorithm like AES with input and output whitening to provide both encryption and authentication. IACBC uses CBC and CBC-MAC.

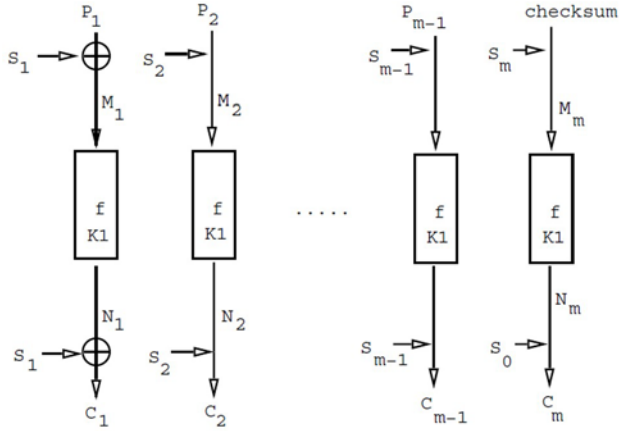

Figure 4: Structure of IAPM[14]

The variations of IAPM-family has many similarities. First, IAPM-family requires a decryption algorithm to decrypt the ciphertext. Second, IAPM-family has a preprocessing capability. Third, IAPM-family requires some memory space to store fixed constants used to whiten input and output. Fourth, IAPM-family produces arbitrary tag length. Fifth, IAPM-family requires one 128 bit of initialization vector. Sixth, IAPM-family is on-line, and has provable security. Last, IAPM-family is patented by a third party.

The variations of IAPM-family have different features. First, the parallelizability is different. IAPM and OCB are fully parallelizable in both encryption and authentication at block-level; encryption and authentication can be executed in one block cipher invocation. IACBC is the two-pass EtM mode version of IAPM, so only its encryption part is parallelizable at block-level. Second, the number of keys required is different. IACBC and IAPM require two keys to execute encryption and authentication, but OCB requires only one key.

### 3.3 Others

Some Authenticated Encryption mode submissions are not similar to others in structure. The modes in this group are all two-pass EtM modes. RKC (Random Key Chaining)[17] uses DRBG (Deterministic Random Bit Generator) to generate random keys for each block of message. DRBG takes a secret 440 bit seed as an input. RKC can preprocess DRBG part, and it requires memory space to store the random keys generated. RKC uses AES-256 to encrypt and SHA-256 to authenticate. RKC does not allow 128 bit or 192 bit key for AES. The encryption part is parallelizable in block-level, and the authentication part is parallelizable in bit-level. RKC has 32 byte tag length, and requires a decryption circuit. RKC is an on-line operation with provable security. RKC is patent-free.

IOC (Input and Output Chaining)[18] uses only AES which input is chained to output of next block and output is chained to input of next block. IOC has minimum preprocessing capability. IOC has very low memory requirement, and the length of a tag is same as that of a key. IOC requires two 128 bit initialization vectors, and is not parallelizable. IOC does not require a decryption circuit. IOC is not an on-line operation, but has provable security. IOC is patent-free.

PCFB (Propagating Cipher Feedback Mode)[19] modifies AES-CFB mode of operation. The structure is very similar to CFB. PCFB has a low preprocessing capability. PCFB has low memory requirement, and the length of a tag is same as that of a key. PCFB requires one 128 bit initialization vector, and is not parallelizable. PCFB does not require a decryption circuit. PCFB is not an on-line operation, but has provable security. PCFB is patent-free.

CS (Cipher-State)[20] uses AES only or the combination of AES and SHA for encryption and authentication. CS use a simple LFSR (Linear Feedback Shift Register) as a pseudo-random number generator to whiten the input (plaintext). CS divides AES into two stages to extract an intermediate value, which will be used to authenticate the message. Only LFSR part can be preprocessed. CS requires some memory space to store its intermediate values, and its tag length is 16 ~ 64 bytes. CS requires one initialization vector which length is same as the length of key. Encryption part of CS is parallelizable in block-level, and authentication part is parallelizable in bit-level. CS requires a decryption circuit. CS is an on-line operation with provable security. CS is patent-free.

## 4 Analysis

### 4.1 Summary

All of Authenticated Encryption mode submissions should be analyzed to understand and extract the key features to evaluate the submissions. After the careful and unbiased examination of the submissions, 16

features are extracted; the number of pass, scheme, underlying algorithm, parallelizability, preprocessing, message length, memory requirement, ciphertext expansion (tag length), key, nonce/IV, performance in parallel, performance in serial, decryption circuit, on-line, patent, and provable security. We present the full summary table in Appendix A. The table is summarized to compare the submissions easily. EAX2 is excluded in the table, because EAX2 is simply a general composition paradigm version of EAX.

In Appendix A, block-level parallelizability allow simultaneous execution of all message blocks. Bit-level parallelizability is the parallelizability in a single block, and it depends on the parallelizability of the underlying algorithms like AES or SHA. The performance is measured in number of underlying algorithm invocations. The time complexity of other operations are not counted since the time complexity of other operations are much smaller than the time complexity of AES and SHA. Performance in parallel feature shows the performance of each mode when executed in parallel without preprocessing. The hardware features are not included in the features of Appendix A, because only a few Authenticated Encryption mode submissions explicitly denote their hardware features. (For example, GCM is cost-efficient when implemented in hardware.) Therefore, the hardware features of mode submissions are not able to be compared, unless hardware implementation is done and compared for all mode submissions.

### 4.2 Important features

The most important features are our choice of the most valued features to evaluate Authenticated Encryption mode submissions. We predict that these features will be the evaluation criteria of CAESAR competition. The key features are parallelizability, preprocessing, memory requirement, key, performance, decryption circuit, on-line, patent, and security strength. # of pass and scheme are not selected, because they are less important than parallelizability and other features. Nonce and initialization vector are excluded, because they are included in memory requirement feature. The security strength is selected, because it is the most important feature of the cryptosystem.

### 4.3 Strong candidates

We have analyzed Authenticated Encryption mode submissions to NIST using the features we extracted. The submission to CAESAR is not available publicly. Thus, we have evaluated only the submissions to NIST. We have chosen two strong candidates which definitely have essential features to survive in CAESAR competition. The first one is GCM. GCM satisfies most of the important features we have chosen. The performance of GCM is not faster than other submissions, but GCM is very cost-efficient when implemented in hardware. The performance of GCM can be improved if more memory space is used. The second one is OCB, which also satisfies most of the features we have chosen. OCB is patented by IBM, but the performance of OCB is exceptional. The performance can be improved if the constant calculation is preprocessed and the constants are saved in the memory. The submissions to CAESAR must be competitive compared to GCM and OCB to survive the competition. The features apply to both algorithm and mode submissions.

Note that the choice of strong candidates is based on the best of our personal analysis.

## 5 Future work

We have extracted the important features and evaluated Authenticated Encryption mode submissions to NIST. We will use the key features presented in this paper to devise our own Authenticated Encryption mode, and submit to both NIST and CAESAR. NIST has no deadline for submissions, and the due date for CAESAR competition is March 15th, 2014.

## 6 Conclusion

All 14 modes of Authenticated Encryption submitted to NIST are surveyed. The pros and cons of the submissions are summarized in Appendix A, and divided the submissions into three groups according to their structural relationships. In the analysis, the necessary features for the evaluation of the submissions are extracted. The most important features required for the submissions to survive CAESAR competition are selected. The features will be used as the guide to devise our own mode of

Authenticated Encryption. Our own mode of Authenticated Encryption will be submitted to both NIST and CAESAR for consideration. Furthermore, two strong candidates among the submissions to NIST are selected using the most important features. Two strong candidates are OCB and GCM.

This paper helps other researchers to understand Authenticated Encryption easily, and provides a short list of necessary features the submissions must satisfy to win the competition.

## Acknowledgement

This research was supported by the KUSTAR-KAIST Institute, Korea, under the R&D program supervised by the KAIST and funded by the MSIP (Ministry of Science, ICT & Future Planning), Korea in the ICT R&D Program 2013.

## References

[1] 44 USC 3542 – Definitions, Legal Information Institute, Cornell University of Law School, http://www.law.cornell.edu/uscode/text/44/3542, Accessed : November 27th, 2013.

[2] Mihir Bellare and Chanathip Namprempre, "Authenticated encryption: Relations among notions and analysis of the generic composition paradigm," *Advances in Cryptology—ASIACRYPT 2000,* Springer Berlin Heidelberg, 2000, 531-545.

[3] Encryption modes development - NIST, http://csrc.nist.gov/groups/ST/toolkit/BCM/modes_development.html, Accessed : November 21th, 2013.

[4] CAESAR - Competition for Authenticated Encryption: Security, Applicability, and Robustness, http://competitions.cr.yp.to/caesar.html, Accessed : November 21th, 2013.

[5] CRYPTREC, http://www.cryptrec.go.jp/english/, Accessed : December 12th, 2013.

[6] Mihir Bellare and Phillip Rogaway, "Encode-then-encipher encryption: How to exploit nonces or redundancy in plaintexts for efficient cryptography," *Advances in Cryptology—ASIACRYPT 2000,* Springer Berlin Heidelberg, 2000, 317-330.

[7] Jee Hea An and Mihir Bellare, "Does encryption with redundancy provide authenticity?," *Advances in Cryptology—EUROCRYPT 2001,* Springer Berlin Heidelberg, 2001, 512-528.

[8] Doug Whiting, Russ Housley, and Niels Ferguson, "Counter with CBC-MAC (CCM)," *AES modes of operations,* 2002.

[9] Mihir Bellare, Phillip Rogaway, and David Wagner, "A conventional authenticated-encryption mode," *AES modes of operations,* April, 2003.

[10] Tadayoshi Kohno, John Viega, and Doug Whiting, "CWC: A high-performance conventional authenticated encryption mode," *Fast Software Encryption,* Springer Berlin Heidelberg, 2004.

[11] David McGrew and John Viega, "The Galois/ Counter mode of operation (GCM)," *Submission to NIST, http://csrc.nist.gov/CryptoToolkit/modes/proposedmodes/gcm/gcm-spec.pdf,* 2004.

[12] Phillip Rogaway and Thomas Shrimpton, "The SIV mode of operation for deterministic authenticated-encryption (key wrap) and misuse-resistant nonce-based authenticated-encryption," *Submission to NIST,* 2007.

[13] Avygdor Moise, Edward Beroset, Tom Phinney, and Martin Burns, "EAX' Cipher Mode," *American National Standards Institute (ANSI) C12 SC17 Committee,* 2011.

[14] Charanjit S. Jutla, "Parallelizable Encryption Mode with Almost Free Message Integrity," *Submission to NIST,* 2000.

[15] Charanjit S. Jutla, "Encryption modes with almost free message integrity," *Advances in Cryptology—EUROCRYPT 2001*, Springer Berlin Heidelberg, 2001, 529-544.

[16] Phillip Rogaway, Mihir Bellare, and John Black, "OCB: A block-cipher mode of operation for efficient authenticated encryption," *ACM Transactions on Information and System Security (TISSEC)*, 6.3, 2003, 365-403.

[17] Puneet Kumar Kaushal, Rajeev Sobti, and G. Geetha, "Random Key Chaining (RKC): AES Mode of Operation," *International Journal of Applied Information Systems*, 1, 2012, 39-45.

[18] Francisco Recacha, "IOC: The Most Lightweight Authenticated Encryption Mode?," *Submission to NIST*, 2013.

[19] Henrick Hellström, "Propagating Cipher Feedback mode," *2nd NIST Modes of Operation Workshop,* August, 2001.

[20] Richard C. Schroeppel, W. Erik Anderson, Cheryl L. Beaver, Timothy J. Draelos, and Mark D. Torgerson, "Cipher-state (CS) mode of operation for AES," *Submission to NIST's Computer Security Resource Center (CSRC) forum on modes of operation for symmetric key block ciphers*, 2004.

## Appendix A : Comparison of Authenticated Encryption mode submissions to NIST

| Features | RKC | CWC | CCM | GCM | EAX | EAX' |
|---|---|---|---|---|---|---|
| # of pass | 2-pass | 2-pass | 2-pass | 2-pass | 2-pass | 2-pass |
| Scheme | EtM | EtM | EtM | EtM | EtM | EtM |
| Underlying algorithm | DRBG, AES, SHA | AES-CTR, C-Hash | AES-CTR, CBC-MAC | AES-CTR, G-Hash | AES-CTR, OMAC | AES-CTR, CMAC |
| Parallelizability | E(B), A(b) | E(B), A(b) | E(B) | E(B), A(b) | E(B) | E(B) |
| Preprocessing | M * DRBG | (M + 1) * AES | M * AES | (M + 2) * AES + 2 * G-Hash | (M + H + N) * AES | (M + N) * AES |
| Message length | $< 2^{64}-1$ | $< 128 * 2^{31}$ | $< 2^{61}$ | $< 2^{39}-256 + 2^{64}$ | Arbitrary | Arbitrary |
| Memory requirement | M * \|Key\| | Small constant | Low | Small constant | Small constant | Small constant |
| Tag length (byte) | 32 | Minimum | 4~16 | 8~16 | < \|Key\| | < \|Key\| |
| Key (bit) | 256 | 128/192/256 | 128/192/256 | 128/192/256 | 128/192/256 | 128/192/256 |
| Unkeyed parameters | 440bit Seed | 88bit Nonce | 128bit Nonce, Counter, and arbitrary AD | 64/96bit IV | Arbitrary Nonce | Arbitrary Nonce |
| Performance – Parallel | M * DRBG + AES + SHA | 3 * AES + C-Hash | (M + 2) * AES | 2 * AES + (M + 3) * G-Hash | (H + N + M + 1) * AES | (N + M + 1) * AES |
| Performance – Serial | M * DRBG + M * AES + SHA | (M + 3) * AES + C-Hash | (2M + 1) * AES | (M + 2) * AES + (M + 3) * G-Hash | (2M + H + N) * AES | (2M + N) * AES |
| Decryption required | Yes | No | No | No | No | No |
| On-line | Yes | Yes | No | Yes | Yes | Yes |
| Patent | No | No | No | No | No | No |
| Provable security | Yes | Yes | Yes | Yes | Yes | Yes |

| Features | IACBC | IAPM | OCB | IOC | PCFB | CS | SIV |
|---|---|---|---|---|---|---|---|
| # of pass | 2-pass | 1-pass | 1-pass | 2-pass | 2-pass | 2-pass | 2-pass |
| Scheme | EtM | E&M | E&M | EtM | EtM | EtM | MtE |
| Underlying algorithm | AES-CBC CBC-MAC | AES | AES | AES | AES-CFB | AES or AES, SHA | AES-CTR, CMAC |
| Parallelizability | E(log(B+1)) | E(B), A(B) | E(B), A(B) | None | None | E(B), A(b) | E(B) |
| Preprocessing | log(M+1) * AES | AES + α | 2 * AES + ϒ | Minimum | 1 * AES | LFSR | (H + M + 1) * CMAC |
| Message length | Arbitrary | Arbitrary | Arbitrary | Arbitrary | Arbitrary | Arbitrary | $< n2^{31}$ or $n2^{63}$ |
| Memory requirement | Modest | Modest | Modest | Low | Low | Modest | Low |
| Tag length (byte) | Minimum | Minimum | Minimum | \|Key\| | \|Key\| | 16 ~ 64 | 8 ~ 16 |
| Key (bit) | 128/192/256 (Two keys) | 128/192/256 (Two keys) | 128/192/256 | 128/192/256 | 128 | 128/192/256 | 128/192/256 (Two keys) |
| Unkeyed parameters | 128bit IV | 128bit IV | \|Key\| IV | 2 * 128bit IV | 128bit IV | \|Key\| IV | Arbitrary Nonce |
| Performance – Parallel | (M + 4) * AES | 2 * AES | 3 * AES | N/A | N/A | LFSR + AES + SHA | (H + M + 1) * CMAC + AES |
| Performance – Serial | (M + log(M+1) + 3) * AES | (M + 2) * AES | (M + 2) * AES | (M + 1) * AES | M * AES | LFSR + M * AES + SHA | (H + M + 1) * CMAC + M * AES |
| Decryption required | Yes | Yes | Yes | No | No | Yes | No |
| On-line | Yes | Yes | Yes | No | No | Yes | No |
| Patent | Yes | Yes | Yes | No | No | No | No |
| Provable security | Yes | Yes | Yes | Yes | Yes | Yes | Yes |

E : Encryption, A : Authentication, (B) : Block-level, (b) : Bit-level.

E(B) : Encryption is parallelizable in block-level, A(B) : Authentication is parallelizable in block-level

A(b) : Authentication is parallelizable in bit-level

IV : Initialization Vector, AD : additional Authentication Data, α, ϒ : complexity of computing constants

M : # of message blocks, H : # of header blocks, N : # of nonce blocks

C-hash : Universal hashing using Carter-Wegman, G-hash : C-hash under binary Galois field