

Message Authentication for Authenticated Encryption Scheme

Rafał Gliwa, Wojciech Oszywa,
Military Communication Institute,
05-130 Zegrze Południowe, Poland,
{r.gliwa, w.oszywa}@wil.waw.pl

Abstract. Data encryption need not prevent an adversary from being able to make the receiver recover data different from that which the sender had intended. It means, that privacy does not imply authenticity and authentication should be a strongly desirable property of any symmetric encryption scheme. One of methods for combining message encryption and authentication are authenticated encryption modes of operation, which clearly specify how to achieve both privacy and authenticity simultaneously. In this paper we focus on authentication component of AE mode and we present algorithms designed for the specific purpose of message authentication i.e.: Message Authentication Codes (MACs) based on CBC mode, MACs based on cryptographic hash functions and MACs based on universal hashing. We try to choose the best one for use as the underlying component in authenticated encryption scheme.

Keywords: authenticated encryption, message authentication, Message Authentication Code, mode of operation.

We know, we have to encrypt data to provide privacy, but quite often it is suggested to encrypt as a way to provide data authenticity, too. Suppose, for example, that the Sender S transmits an encrypted message M_{100} which indicates that the Receiver should please transfer \$100 from the checking account of S to the checking account of some other party. The adversary A wants to change the amount from the \$100 to \$900. She does not know the key K , so she cannot just encrypt M_{900} on her own. It seems, the privacy of C_{100} already rules out that C_{100} can be profitably tampered with. To see the flaws let's look at a counter-example [12]. If we encrypt M_{100} using a one time pad, then all the adversary has to do is to XOR the byte of the ciphertext C_{100} which encodes the character "1" with the XOR of the bytes which encode "1" and "9". That is, when we one-time pad encrypt, the privacy of the transmission does not make it difficult for the adversary to tamper with ciphertext so as to produce related ciphertexts. The fact that data is encrypted need not prevent an adversary from being able to make the receiver recover data different from that which the sender had intended. Encrypting a message was never an appropriate approach for protecting its authenticity. Good cryptographic design is goal-oriented. Message authentication oriented designs are Message Authentication Code (MAC) algorithms [15].

Message authentication allows one party, the Sender, to send a message to another party, the Receiver, in such a way that if the message is modified en route, then the Receiver will almost certainly detect this. Message authentication is also called "data

origin authentication", since it corroborates the source of origin for each message. Message authentication protects also the „integrity” of messages, ensuring that each, that is received and deemed acceptable, is arriving in the same condition that it was sent out with no bits inserted, missing or modified. A MAC is a function that takes an input of arbitrary length and produces an output of a fixed length. Contrary to hash functions, the computation of a MAC depends on a secret key K . In practical applications this key has to be shared between two parties, the Sender S and the Receiver R , so MACs are used in a symmetric settings (contrary to digital signatures, which are used for authentication in asymmetric settings). In order to protect a message M , the Sender applies the MAC algorithm to M and appends the resulting string Tag to the message. The Adversary A controls the channel, so we cannot be sure that M and Tag reach their intended destination. The Receiver gets M' and Tag' and applies a verification function to K , M' and Tag' to decide if M' should be regarded as the source message M , or as the adversary's creation. An active adversary can modify the message, but as she does not know the secret key K , she cannot predict the Tag value for the modified message.

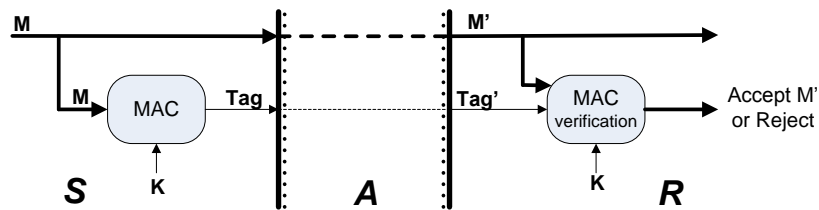


Fig. 1 Message Authentication Code (MAC) scheme.

The following definition for Message Authentication Code was given by Preneel in [15]. A MAC is a function h satisfying the following conditions:

1. The description of h must be publicly known and the only secret information lies in the key.
2. The argument M can be of arbitrary length and the result $h_K(M)$ has a fixed length of n bits.
3. Given h , M and K , the computation of $h_K(M)$ must be “easy”.
4. Given h and M , it is “hard” to determine $h_K(M)$ with a probability of success “significantly higher” than $1/2^n$. Even when a large set of pairs $\{M_i, h_K(M_i)\}$ is known, where the M_i have been selected by the opponent, it is “hard” to determine the key K or to compute $h_K(M')$ for any $M' \neq M_i$.

There are a few constructions that are designed for the specific purpose of message authentication. One of the first approaches for designing a MAC was to build it on the top of an existing block cipher E and then proceed with a common mode of operation. Such MAC is the CBC-MAC, which is a well studied method to generate a message authentication code based on the Cipher Block Chaining mode [12].

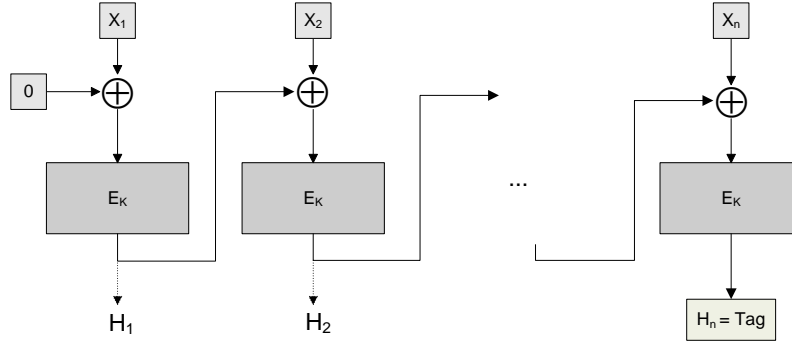


Fig. 2 CBC-MAC scheme.

The MAC key is used as cipher key in each step of the iteration, and the message block to be processed in the current step serves as plaintext input to the cipher, after being added bit by bit to the ciphertext output from the previous step:

$$H_i = E_K(X_i); \quad (1)$$

$$H_i = E_K(X_i \oplus H_{i-1}), \quad (2 \leq i \leq n). \quad (2)$$

Here we assume that the message X (after padding) is divided into blocks X_1, \dots, X_n of lengths appropriate for the block cipher used. E_K denotes encryption with secret key K and H_n forms the output of the MAC algorithm. The inherited CBC chaining dependency leads to a general disability of the CBC-MAC to process messages in parallel. There is also no possibility for preprocessing, because it is necessary to know the result of processing the previous block of message in order to process the next block. Furthermore, CBC-MAC is insecure for arbitrary long messages [2]. Several more secure variations of the scheme exist however e.g. EMAC [4], XCBC [4], OMAC [10], RIPE-MAC [5]. Eventually, one of CBC-MAC like construction was adapted to the AES block cipher and NIST published CMAC [13] as the recommended block cipher mode for message authentication.

An alternative type of MAC construction are message authentication codes based on a cryptographic hash function. This is a common approach because these MACs are usually faster than MACs based on a block cipher. HMAC [1] is a nested construction that computes a MAC for an underlying hash function h , message X and secret key K , as follows (*opad* and *ipad* are constant values):

$$\text{HMAC}(K, X) = h(K \oplus \text{opad} \parallel h(K \oplus \text{ipad} \parallel X)). \quad (3)$$

The objective of HMAC is to use, with no modification, hash function for its MAC construction. The security of HMAC is based entirely on the underlying hash function, meaning that a weakness in the MAC would only appear if the hash function has not enough cryptographic strength. HMAC is generally much faster than CBC-MAC construction and the reason behind it is that cryptographic hash functions are significantly faster than the multiple block cipher operations used in the latter.

Message Authentication Codes given above use a single cryptographic primitive for their constructions, either block cipher or cryptographic hash function. Carter and Wegman suggested different approach for designing a MAC, namely how to MAC using Universal Hash Function Families [6][7].

Definition 1. Fix a domain D and range R . A finite multiset of hash functions $H=\{h: D\rightarrow R\}$ is said to be Universal if for every $x, y \in D$, where $x \neq y$, $Pr_{h \in H} [h(x)=h(y)] \leq 1/|R|$.

Their idea was quite novel: instead of applying some cryptographic primitive to the message X to be MACed, they would first hash X down to a smaller size using a hash function drawn from a Universal Hash Function Family, which had only a combinatorial property (rather than a cryptographic one). Then they would apply a cryptographic primitive one-time pad encryption to the smaller resulting string.

$$Tag_K(X, N) = h_{K1}(X) \oplus f_{K2}(N), \quad (4)$$

where:

- Tag - authentication tag,
- X - message to be authenticated,
- N - nonce (number used once);
- K - keys space, $K1, K2 \in K$,
- h_{K1} - function drawn from a Universal Hash Function Family,
- f_{K2} - block cipher encryption.

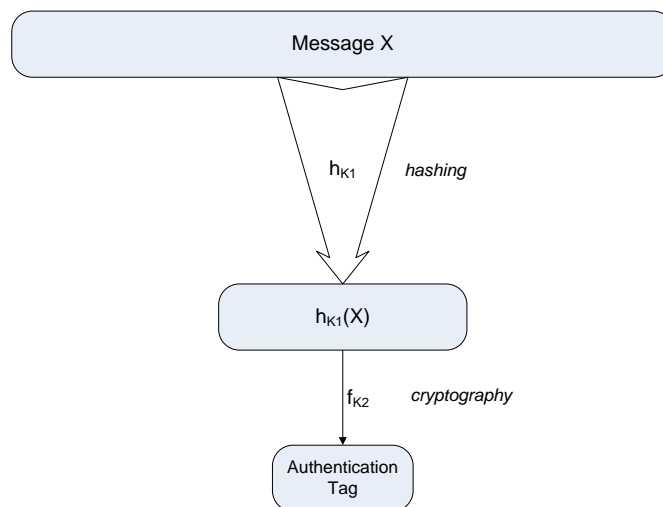


Fig. 3 Carter-Wegman MAC.

The combinatoric property of the universal hash function family is often not difficult to prove, and it can be shown that the security of the resulting MAC scheme depends on the security of the cipher that is used. The universal hash is quite fast on most modern processors, but in the search for efficiency researchers have discovered

that even faster methods are possible if we relax slightly the requirement that the collision probability be $1/R$. This led to the notion of Almost Universal Hash Families in which we allow the collision probability to be some $\varepsilon \geq 1/R$.

Definition 2. Let ε be a positive number. Fix a domain D and range R . A finite multiset of hash functions $H = \{h: D \rightarrow R\}$ is said to be Almost Universal if for every $x, y \in D$, where $x \neq y$, $\Pr_{h \in H} [h(x) = h(y)] \leq \varepsilon$.

Universal hash function families based MACs include, but are not limited to UMAC [3], cryptographic CRC [9], bucket hashing [11] or MMH [8]. An extensive study for the performance of several universal hash functions for MACs is given by Nevelsteen and Preneel in [11].

Recently, lots of efforts are put into area of authenticated encryption (AE), i.e. into providing both privacy and authenticity of the message simultaneously. AE scheme combines encryption component with authentication component in a secure way. The most straightforward option is to calculate the MAC, append it to the information and subsequently encrypt the new message. An alternative is to omit the encryption of the MAC. The third solution is to calculate the MAC on the encrypted message, then the advantage is that the authenticity can be verified by a receiver without knowing the plaintext (with no need to decrypt if a message is not valid). Particularly desirable AE scheme properties are high level of security and high speed of message processing what is associated with parallelizability requirement. Therefore, the designer of AE scheme should carefully consider how the choice of encryption and authentication components will affect these characteristics.

As we have already found we have three candidate constructions for authentication component. MAC algorithms built on block ciphers are distinguished by characteristics inherent in CBC mode idea i.e. chaining encrypted blocks. Therefore, a common feature of all CBC-MAC variants is lack of parallelism. Significant improvement in performance over the CBC-MAC show MAC algorithms based on cryptographic hash function eg. HMAC. Indeed, hash functions in general are significantly faster than the multiple block cipher operations. Unfortunately, iterative nature of cryptographic hash based MAC constructions makes impossible parallel computations as well. If it was not considered as a problem in computation in the past, currently and in the future this shortcoming turns out to be a significant drawback. Inability of parallel computations influences directly on reducing efficiency of message processing, which is particularly undesirable feature.

Message authentication algorithms based on Carter-Wegman construction are devoid of such disadvantage. In their case, the efficiency problem of fast message authentication has been reduced to fast universal hashing. Therefore, Carter-Wegman MAC schemes are the fastest MACs around. The unquestionable advantage of Carter-Wegman MAC constructions is that their security is unconditional (one-time pad operation on the hash value). Taking into account that recent findings of internal collisions in cryptographic hash functions, such as SHA-2, seem to endanger the security of these functions as well as cryptographic hash based MAC algorithms, it is not surprising that universal hashing techniques gather popularity and they are more and more widely used and explored. We can clearly state that universal hashing approach seem to remain at present most perspective variant for the construction of secure and efficient MAC algorithms.

Thus, universal hash based MAC seem to be also the most appropriate construction for use as an underlying message authentication component in an authenticated encryption scheme. Application of this solution in combination with encryption component has one more significant advantage - there is no need to implement cryptographic hash function. Both for message encryption and authentication block cipher algorithm is used only. It significantly reduces the required hardware resources. When evaluating the effectiveness of authenticated encryption mode of operation the total number of cipher invocation is taken into account. From this point of view, application of universal hash based MAC component in AE scheme makes, that providing both message privacy and authentication requires only a little more (even only one more) cipher invocations than providing message privacy only. The rest of calculations are just simple and fast combinatorial operations. Combining encryption component and universal hash based MAC component has already been reflected in practice. First AE schemes built within CBC-MAC component did not allow to reach speeds of more than 2Gb/s. Therefore AE schemes based on universal hash based MAC component developed at a later time, which allowed for increasing that limit to the speed of even 10 Gb/s. The best known example of combining encryption component and universal hash based MAC component is two pass authenticated encryption mode GCM [14].

BIBLIOGRAPHY

1. Bellare, M., Canetti, R., Krawczyk H., Keying Hash Functions for Message Authentication, *Advances in Cryptology, Proc. Crypto '96*, LNCS 1109, Springer-Verlag, 1996.
2. Bellare, M., Kilian, J., Rogaway, P., The security of Cipher Block Chaining, *Proc. Crypto'94*, LNCS 839, Springer-Verlag, 1994.
3. Black, J., Halevi, S., Krawczyk, H., Krovetz, T., Rogaway, P., UMAC: Fast and Secure Message Authentication, *Proc. Crypto '99*, LNCS 1666, Springer-Verlag, 1999.
4. Black, J., Rogaway, P., CBC MACs for Arbitrary-Length Messages: The Three-Key Constructions, *Proc. Crypto '00*, LNCS 1880, Springer-Verlag, 2000.
5. Bosselaers, Preneel, B., RIPE, Integrity Primitives for Secure Information Systems, Final Report of RACE Integrity Primitives Evaluation, 1992.
6. Carter, L., Wegman, M. N., Universal Classes of Hash Functions, *Journal of CSS*, 1979.
7. Carter, L., Wegman, M. N., New hash functions and their use in authentication and set equality, *Journal of Computer and System Sciences*, vol. 22, 1981.
8. Halevi, S., Krawczyk, H., MMH: Software Message Authentication in the Gbit/Second Rates, *Fast Software Encryption, Proc. FSE'97*, LNCS 1267, 1997.
9. Krawczyk, H., LFSR-based Hashing and Authentication. *Proc. CRYPTO'94*, LNCS 839, Springer-Verlag, 1994.
10. Iwata, T., Kurosawa, K., OMAC: One-Key CBC MAC, *Proc. FSE'03*, LNCS 2887, 2003.
11. Nevelsteen, W., Preneel, B., Software Performance of Universal Hash Functions, *Proc. Eurocrypt' 99*, LNCS 1592, 1999.
12. NIST Special Publication 800-38A, Recommendation for Block Cipher Modes of Operation - Methods and Techniques, December 2001.
13. NIST Special Publication 800-38B, Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, May 2005.
14. NIST Special Publication 800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, November 2007.
15. Preneel, B., Analysis and design of cryptographic hash functions. PhD thesis, Katholieke Universiteit Leuven, Belgium, 2003.