Groupe de travail Réseau
**Request for Comments : 3566**
Catégorie : En cours de normalisation
Traduction Claude Brière de L'Isle

S. Frankel, NIST
H. Herbert, Intel
septembre 2003

# Algorithme AES-XCBC-MAC-96 et son utilisation avec IPsec



**Résumé**
Le code d'authentification de message (MAC, *Message Authentication Code*) est une fonction de hachage unidirectionnelle dépendant d'une clé. Une façon populaire de construire un algorithme de MAC est d'utiliser un chiffrement de bloc en conjonction avec le mode de fonctionnement en chaînage de bloc de chiffrement (CBC, *Cipher-Block-Chaining* ). L'algorithme CBC-MAC classique, bien que sûr pour les messages d'une longueur fixe présélectionnée, s'est révélé non sûr sur des messages de longueur variable tels que du type qu'on trouve dans les datagrammes IP courants. Le présent mémoire spécifie l'utilisation de AES en mode CBC avec un ensemble d'extensions pour surmonter cette limitation. Ce nouvel algorithme est appelé AES-XCBC-MAC-96.

## Table des matières



## 1. Introduction

L'authentification de message assure la protection de l'intégrité des données et l'authentification de l'origine des données par rapport à la source originale du message. Un code d'authentification de message MAC, *Message Authentication Code*) est une fonction de hachage unidirectionnelle dépendant d'une clé. Une façon populaire de construire un algorithme de MAC est d'utiliser un chiffrement de bloc en conjonction avec le mode de fonctionnement en chaînage de bloc de chiffrement (CBC, *Cipher-Block-Chaining*). L'algorithme CBC-MAC classique, bien que sûr pour les messages de longueur fixe présélectionnée [CBC-MAC-2], s'est révélé non sûr pour les messages de longueurs variables tels que du type qu'on trouve dans les datagrammes IP courants [CBC-MAC-2, section 5]. En fait, il est trivial de produire des falsifications pour un second message connaissant le MAC d'un message précédent [HANDBOOK, section 9.62, p. 354]

Le présent mémoire spécifie l'utilisation de [AES] en mode CBC [MODES] avec un ensemble d'extensions [XCBC-MAC-

1] pour surmonter cette limitation. Ce nouvel algorithme est appelé AES-XCBC-MAC-96. L'utilisation du chiffrement de bloc AES, avec sa taille de bloc accrue (128 bits) et sa longueur de clé augmentée (128 bits) donne au nouvel algorithme la capacité de résister aux avancées continuelles des techniques de cryptanalyse et des capacités de calcul. AES-XCBC-MAC-96 est utilisé comme mécanisme d'authentification dans le contexte des protocoles IPsec d'encapsulation de charge utile de sécurité (ESP, *Encapsulating Security Payload*) et d'en-tête d'authentification (AH, *Authentication Header*). Pour plus d'informations sur ESP, se référer à la [RFC2406] et à la [RFC2411]. Pour plus d'informations sur AH, se référer aux [RFC2402] et [RFC2411].

Le but de AES-XCBC-MAC-96 est d'assurer que le datagramme est authentique et ne peut pas être modifié dans le transit. L'intégrité et l'authentification d'origine des données telles que fournies par AES-XCBC-MAC-96 dépendent de la portée de la distribution de la clé secrète. Si la clé n'est connue que de la source et de la destination, cet algorithme fournira à la fois l'authentification de l'origine des données et la protection de l'intégrité pour les datagrammes envoyés entre les deux parties. De plus, seule une partie ayant la clé identique peut vérifier le hachage.

## 2. Spécification des exigences

Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT" et "FACULTATIF" dans ce document sont à interpréter comme décrit dans la [RFC2119], BCP 14.

## 3. CBC-MAC de base avec bourrage 10* obligatoire

CBC-MAC utilise un chiffrement de bloc pour le chiffrement ; le chiffrement de bloc transforme b bits de texte en clair en b bits de texte chiffré. Le CBC-MAC de base [CBC-MAC-1], [CBC-MAC-2] avec bourrage 10* obligatoire sur un chiffrement de bloc de b bits est calculé comme suit pour un message M :

(1) Ajouter un seul bit 1 à M. Puis ajouter le nombre minimum de bits 0 bits à M de sorte que la longueur de M soit un multiple de b. (Note : C'est un schéma de bourrage parmi plusieurs qui peuvent être utilisés pour CBC-MAC. Plusieurs autres sont décrits dans [MODES].)

(2) Casser M en n blocs, M[1] ... M[n], où la taille de bloc des blocs M[1] ... M[n] est de b bits.

(3) Définir E[0] = 0x00000000000000000000000000000000

(4) Pour chaque bloc M[i], où i = 1 ... n: OUX M[i] avec E[i-1], puis chiffrer le résultat avec la clé K, ce qui donne E[i].

(5) E[n] est l'authentifiant à b bits.

Le CBC-MAC de base avec bourrage 10* obligatoire s'est révélé sûr pour les messages jusqu'à (mais non inclus) une longueur fixe présélectionnée, dans laquelle la longueur est un multiple de la taille de bloc. Cet algorithme ne convient pas pour IPsec pour les raisons suivantes :

+ Tout authentifiant IPsec doit être capable de traiter des messages de longueur arbitraire. Cependant, le CBC-MAC de base ne peut pas traiter en toute sécurité les messages qui excèdent la longueur fixe présélectionnée.
+ Pour les messages plus courts que la longueur fixe présélectionnée, le bourrage du message à la longueur fixe présélectionnée peut nécessiter des opérations de chiffrement supplémentaires, ajoutant une charge de calcul inacceptable.

## 4. AES-XCBC-MAC-96

[AES] décrit l'algorithme AES sous-jacent, tandis que [CBC-MAC-1] et [XCBC-MAC-1] décrivent l'algorithme AES-XCBC-MAC.

L'algorithme AES-XCBC-MAC-96 est une variante du CBC-MAC de base avec bourrage 10* obligatoire, cependant, AES-XCBC-MAC-96 est sûr pour les messages de longueur arbitraire. Les calculs de AES-XCBC-MAC-96 exigent de nombreuses opérations de chiffrement ; ce chiffrement DOIT être accomplis en utilisant AES avec une clé de 128 bits. Étant donnée une clé secrète K de 128 bits, AES-XCBC-MAC-96 est calculé comme suit pour un message M qui consiste en n blocs, M[1] ... M[n], dans lesquels la taille de bloc des blocs M[1] ... M[n-1] est 128 bits et la taille de bloc du bloc M[n] est entre 1 et 128 bits:

(1) Déduire trois clés de 128 bits (K1, K2 et K3) de la clé secrète K de 128 bits, comme suit :
K1 = 0x01010101010101010101010101010101 chiffré avec la clé K
K2 = 0x02020202020202020202020202020202 chiffré avec la clé K
K3 = 0x03030303030303030303030303030303 chiffré avec la clé K

(2) Définir E[0] = 0x00000000000000000000000000000000

(3) Pour chaque bloc M[i], où i = 1 ... n-1 : OUX M[i] avec E[i-1], puis chiffrer le résultat avec la clé K1, donnant E[i].

(4) Pour le bloc M[n] :
a) Si la taille de bloc de M[n] est 128 bits : OUX M[n] avec E[n-1] et la clé K2, puis chiffrer le résultat avec la clé K1, donnant E[n].
b) Si la taille de bloc de M[n] est inférieure à 128 bits :
i) Bourrer M[n] avec un seul bit "1", suivi par le nombre de bits "0" (éventuellement aucun) requis pour augmenter la taille de bloc de M[n] jusqu'à 128 bits.
ii) OUX M[n] avec E[n-1] et la clé K3, puis chiffrer le résultat avec la clé K1, donnant E[n].

(5) La valeur de l'authentifiant est les 96 bits les plus à gauche des 128 bits de E[n].

Note 1 : Si M est la chaîne vide, bourrer et chiffrer comme en (4)(b) pour créer M[1] et E[1]. Cela ne sera jamais le cas pour ESP ou AH, mais c'est mentionné par souci d'être complet.

Note2 : [CBC-MAC-1] définit K1 comme suit : K1 = Constant1A chiffré avec la clé K | Constant1B chiffré avec la clé K. Cependant, la seconde opération de chiffrement n'est nécessaire que pour AES-XCBC-MAC avec des clés supérieures à 128 bits ; donc, elle n'est pas incluse dans la définition de AES-XCBC-MAC-96.

La vérification de AES-XCBC-MAC-96 est effectuée comme suit :
À réception de l'authentifiant d'AES-XCBC-MAC-96, la valeur entière de 128 bits est calculée et les 96 premiers bits sont comparés à la valeur mémorisée dans le champ d'authentifiant.

### 4.1 Matériel de clés

AES-XCBC-MAC-96 est un algorithme à clé secrète. Pour l'utilisation avec ESP ou AH, une longueur fixe de clé de 128 bits DOIT être prise en charge. Les longueurs de clé autres que 128 bits NE DOIVENT PAS être acceptées (c'est-à-dire que seules les clés de 128 bits sont à utiliser par AES-XCBC-MAC-96).

AES-XCBC-MAC-96 exige en fait 384 bits de matériel de clés (128 bits pour la taille de clé AES + 2 fois la taille de bloc). Ce matériel de clé peut être fourni au moyen du mécanisme de génération de clé ou bien il peut être généré à partir d'une seule clé de 128 bits. Cette dernière approche a été choisie pour AES-XCBC-MAC-96, car elle est analogue aux autres authentifiants utilisés dans IPsec. La raison pour laquelle AES-XCBC-MAC-96 utilise trois clés est qu'ainsi, la longueur du flux d'entrée n'a pas besoin d'être connue à l'avance. Cela peut être utile pour les systèmes qui font de l'assemblage de gros paquets en une seule passe.

Une fonction pseudo aléatoire forte DOIT être utilisée pour générer la clé de 128 bits requise. Cette clé, avec les trois clés dérivées (K1, K2 et K3) devrait être utilisée pour les seuls besoins spécifiés dans l'algorithme. En particulier, elles ne devraient pas être utilisées comme clés dans un autre assemblage cryptographique. De tels abus invalideraient la sécurité de l'algorithme d'authentification.

Au moment de la rédaction du présent mémoire, il n'y a pas de clé faible spécifiée à utiliser avec AES-XCBC-MAC-96. Cela ne signifie pas qu'il n'existe pas de clés faibles. Si à un moment quelconque, des clés faibles étaient identifiées pour AES-XCBC-MAC-96, l'utilisation de ces clés faibles DEVRA être rejeté et suivi par une demande de remplacement de clés ou d'une nouvelle négociation d'association de sécurité.

La [RFC2401] décrit le mécanisme général d'obtention du matériel de clé lorsque plusieurs clés sont requises pour une seule SA (par exemple, lorsque une SA ESP demande une clé pour la confidentialité et une clé pour l'authentification).

Afin de fournir l'authentification de l'origine des données, le mécanisme de distribution de clés doit assurer que des clés uniques sont allouées et qu'elle ne sont distribuées qu'aux parties qui participent à la communication.

Les attaques courantes ne nécessitent pas de recommander une fréquence spécifique des changements de clé. Cependant, un rafraîchissement périodique des clés est une pratique de sécurité fondamentale qui aide contre les faiblesses potentielles de

la fonction et des clés, réduit les informations disponibles à la cryptanalyse, et limite les dommages résultant de la compromission d'une clé.

### 4.2 Bourrage

AES-XCBC-MAC-96 fonctionne sur des bocs de données de 128 bits. Les exigences de bourrage sont spécifiées dans [CBC-MAC-1] font partie de l'algorithme XCBC. Si on construit AES-XCBC-MAC-96 conformément à [CBC-MAC-1] on n'a pas besoin d'ajouter de bourrage supplémentaire pour ce qui concerne AES-XCBC-MAC-96. Par rapport au "bourrage implicite de paquet" défini dans la [RFC2402], aucun bourrage implicite de paquet n'est requis.

### 4.3 Troncature

AES-XCBC-MAC produit une valeur d'authentifiant de 128 bits. AES-XCBC-MAC-96 est déduit en tronquant cette valeur de 128 bits comme décrit dans la [RFC2104] et vérifié dans [XCBC-MAC-2]. Pour l'utilisation avec ESP ou AH, une valeur tronquée utilisant les 96 premiers bits DOIT être prise en charge. À l'envoi, la valeur tronquée est mémorisée au sein du champ Authentifiant. À réception, la valeur entière de 128 bits est calculée et les 96 premiers bits sont comparés à la valeur mémorisées dans le champ Authentifiant. Aucune autre longueur de valeur d'authentifiant n'est prise en charge par AES-XCBC-MAC-96.

La longueur de 96 bits a été choisie parce qu'elle est la longueur d'authentifiant par défaut spécifiée dans la [RFC2402] et qu'elle satisfait aux exigences de sécurité décrites dans [XCBC-MAC-2].

### 4.4 Interaction avec le mécanisme de chiffrement ESP

Au moment de la rédaction, il n'y a pas de problème connu qui empêche l'utilisation de AES-XCBC-MAC-96 avec aucun algorithme spécifique de chiffrement.

### 4.5 Performances

Pour toute variante de CBC MAC, l'effort de calcul majeur est effectué pour le calcul du chiffrement de bloc sous-jacent. Cet algorithme utilise un nombre minimum d'invocations d'AES, une pour chaque bloc du message ou de ses fractions, d'où il résulte des performances équivalentes au CBC-MAC classique.

L'expansion de clés exige trois opérations de chiffrement AES supplémentaires, mais celles-ci peuvent être effectuées à l'avance pour chaque clé secrète.

### 4.6 Vecteurs d'essai

Ces cas d'essai ont été fournis par John Black, co-auteur de l'algorithme XCBC-MAC, qui les a vérifié avec deux mises en œuvre indépendantes. Toutes les valeurs sont des nombres hexadécimaux.

Cas d'essai n° 1 : AES-XCBC-MAC-96 avec 0-octet en entrée
Clé (K) : 000102030405060708090a0b0c0d0e0f
Message (M) : <chaîne vide>
AES-XCBC-MAC : 75f0251d528ac01c4573dfd584d79f29
AES-XCBC-MAC-96 : 75f0251d528ac01c4573dfd5

Cas d'essai n° 2 : AES-XCBC-MAC-96 avec 3 octets en entrée
Clé (K) : 000102030405060708090a0b0c0d0e0f
Message (M) : 000102
AES-XCBC-MAC : 5b376580ae2f19afe7219ceef172756f
AES-XCBC-MAC-96 : 5b376580ae2f19afe7219cee

Cas d'essai n° 3 : AES-XCBC-MAC-96 avec 16 octets en entrée
Clé (K) : 000102030405060708090a0b0c0d0e0f
Message (M) : 000102030405060708090a0b0c0d0e0f
AES-XCBC-MAC : d2a246fa349b68a79998a4394ff7a263
AES-XCBC-MAC-96 : d2a246fa349b68a79998a439

Cas d'essai n° 4 : AES-XCBC-MAC-96 avec 20 octets en entrée
Clé (K) : 000102030405060708090a0b0c0d0e0f

Message (M) : 000102030405060708090a0b0c0d0e0f10111213
AES-XCBC-MAC : 47f51b4564966215b8985c63055ed308
AES-XCBC-MAC-96 : 47f51b4564966215b8985c63

Cas d'essai n° 5 : AES-XCBC-MAC-96 avec 32 octets en entrée
Clé (K) : 000102030405060708090a0b0c0d0e0f
Message (M) : 000102030405060708090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f
AES-XCBC-MAC : f54f0ec8d2b9f3d36807734bd5283fd4
AES-XCBC-MAC-96 : f54f0ec8d2b9f3d36807734b

Cas d'essai n° 6 : AES-XCBC-MAC-96 avec 34 octets en entrée
Clé (K) : 000102030405060708090a0b0c0d0e0f
Message (M) : 000102030405060708090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f2021
AES-XCBC-MAC : becbb3bccdb518a30677d5481fb6b4d8
AES-XCBC-MAC-96 : becbb3bccdb518a30677d548

Cas d'essai n° 7 : AES-XCBC-MAC-96 avec 1000 octets en entrée
Clé (K) : 000102030405060708090a0b0c0d0e0f
Message (M) : 0000000000000000000 ... 0000000000000000000[1000 octets]
AES-XCBC-MAC : f0dafee895db30253761103b5d84528f
AES-XCBC-MAC-96 : f0dafee895db30253761103b

## 5. Considérations pour la sécurité

La sécurité fournie par AES-XCBC-MAC-96 se fonde sur la force de AES. Au moment de la rédaction de ce mémoire, il n'existe pas d'attaque cryptographique pratique contre AES ou AES-XCBC-MAC-96.

Comme c'est vrai de tout algorithme cryptographique, une partie de sa force réside dans la correction de la mise en œuvre de l'algorithme, dans la sécurité du mécanisme de gestion de clés et de sa mise en œuvre, la force de la clé secrète associée, et dans la correction de la mise en œuvre de tous les systèmes participants. Le présent document contient des vecteurs d'essai pour aider à vérifier la correction du code d'AES-XCBC-MAC-96.

## 6. Considérations relatives à l'IANA

L'IANA a alloué l'identifiant de transformation AH 9 à AH_AES-XCBC-MAC.
L'IANA a alloué la valeur d'algorithme d'authentification AH/ESP 9 à AES-XCBC-MAC.

## 7. Déclaration de droits de propriété intellectuelle

L'IETF ne prend pas position sur la validité et la portée de tout droit de propriété intellectuelle ou autres droits qui pourrait être revendiqués au titre de la mise en œuvre ou l'utilisation de la technologie décrite dans le présent document ou sur la mesure dans laquelle toute licence sur de tels droits pourrait être ou n'être pas disponible ; pas plus qu'elle ne prétend avoir accompli aucun effort pour identifier de tels droits. Les informations sur les procédures de l'ISOC au sujet des droits dans les documents de l'ISOC figurent dans les BCP 78 et BCP 79.

Des copies des dépôts d'IPR faites au secrétariat de l'IETF et toutes assurances de disponibilité de licences, ou le résultat de tentatives faites pour obtenir une licence ou permission générale d'utilisation de tels droits de propriété par ceux qui mettent en œuvre ou utilisent la présente spécification peuvent être obtenues sur répertoire en ligne des IPR de l'IETF à http://www.ietf.org/ipr .

L'IETF invite toute partie intéressée à porter son attention sur tous copyrights, licences ou applications de licence, ou autres droits de propriété qui pourraient couvrir les technologies qui peuvent être nécessaires pour mettre en œuvre la présente norme. Prière d'adresser les informations à l'IETF à ietf- ipr@ietf.org.

## 8. Remerciements

Des portions de ce texte sont directement tirées de [RFC2404]. Merci aux auteurs de XCBC-MAC pour leurs conseils

experts et leur rapide réponse à nos interrogations : à Phil Rogaway pour nous avoir fourni les valeurs des constantes de XCBC-MAC, et à John Black pour les corrections détaillées aux spécifications de l'algorithme et pour la fourniture des cas d'essai. Merci aussi à Andrew Krywaniuk pour avoir insisté pour (et avoir fourni les texte) le motif de l'approche des trois clés.

## 9. Références

### 9.1 Références normatives

[AES] NIST, FIPS PUB 197, "Advanced Encryption Standard (AES)," novembre 2001. http://csrc.nist.gov/publications/fips/fips197/fips-197 {ps, pdf}

[CBC-MAC-1] Black, J. and P. Rogaway, "CBC MACs for Arbitrary-Length Messages: The Three-Key Constructions," dans M. Bellare, éditeur, Advances in Cryptology -- CRYPTO '00, volume 1880 de Lecture Notes dans Computer Science, p. 0197, août 2000, Springer-Verlag. http://www.cs.ucdavis.edu/~rogaway/papers/3k.ps

[RFC2119] S. Bradner, "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997.

[RFC2402] S. Kent et R. Atkinson, "En-tête d'authentification IP", novembre 1998. (*Obsolète, voir* RFC4302, 4305)

[RFC2406] S. Kent et R. Atkinson, "Encapsulation de charge utile de sécurité IP (ESP)", novembre 1998. *(Obsolète, voir RFC4303)*

[XCBC-MAC-1] Black, J. and P. Rogaway, "A Suggestion for Handling Arbitrary-Length Messages with the CBC MAC," NIST Second Modes of Operation Workshop, août 2001. http://csrc.nist.gov/encryption/modes/proposedmodes/xcbc-mac/xcbc-mac-spec.pdf

### 9.2 Références pour information

[CBC-MAC-2] Bellare, M., J. Kilian and P. Rogaway, "The Security of the Cipher Block Chaining Message Authentication Code," Journal of Computer and System Sciences (JCSS), Vol. 61, n° 3, décembre 2000, pp. 362-399. http://www.cse.ucsd.edu/users/mihir/papers/cbc {ps, pdf}

[HANDBOOK] Menezes, A., P. Van Oorschot and S. Vanstone, "Handbook of Applied Cryptography", CRC Press, 1997.

[MODES] Dworkin, M., "Recommendation for Block Cipher Modes of Operation: Methods and Techniques," NIST Special Publication 800-38A, décembre 2001. http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf

[RFC2026] S. Bradner, "Le processus de normalisation de l'Internet -- Révision 3", (BCP0009) octobre 1996. *(Remplace* RFC1602, RFC1871) (*MàJ par* RFC3667, RFC3668, RFC3932, RFC3979, RFC3978, RFC5378, RFC6410)

[RFC2104] H. Krawczyk, M. Bellare et R. Canetti, "HMAC : Hachage de clés pour l'authentification de message", février 1997.

[RFC2401] S. Kent et R. Atkinson, "Architecture de sécurité pour le protocole Internet", novembre 1998. (*Obsolète, voir* RFC4301)

[RFC2404] C. Madson, R. Glenn, "Utilisation de HMAC-SHA-1-96 au sein d'ESP et d'AH", novembre 1998. *(P.S.)*

[RFC2411] R. Thayer, N. Doraswamy, R. Glenn, "Feuille de route pour la sécurité sur IP", novembre 1998. *(Remplacée par* RFC6071*)*

[XCBC-MAC-2] Rogaway, Phil, communications privées, octobre 2001.

## 10. Adresse des auteurs

Sheila Frankel
NIST - National Institute of Standards and Technology
820 West Diamond Ave.
Room 677
Gaithersburg, MD 20899
téléphone : +1 (301) 975-3297
mél : sheila.frankel@nist.gov

Howard C. Herbert
Intel Corporation
Lan Access Division
5000 West Chandler Blvd.
MS-CH7-404
Chandler, Arizona 85226
téléphone : +1 (480) 554-3116
mél : howard.c.herbert@intel.com

## 11. Déclaration complète de droits de reproduction



## Remerciement

Le financement de la fonction d'édition des RFC est actuellement fourni par l'Internet Society.