Groupe de travail Réseau
Request for Comments : 3610
Catégorie : Information
Traduction Claude Brière de L'Isle

D. Whiting, Hifn
R. Housley, Vigil Security
N. Ferguson, MacFergus
septembre 2003

Compteur avec CBC-MAC (CCM)

Résumé
Compteur avec CBC-MAC (CCM, *Counter with CBC-MAC*) est un mode de chiffrement de bloc de chiffrement authentifié générique. CCM est défini pour être utilisé avec des chiffrements de bloc de 128 bits, comme la norme de chiffrement évolué (AES).

Table des matières

1. Introduction

Compteur avec CBC-MAC (CCM, *Counter with CBC-MAC*) est un mode de chiffrement de bloc de chiffrement authentifié générique. CCM n'est défini que pour l'utilisation avec des chiffrements de bloc de 128 bits, comme [AES]. Les principes de conception de CCM peuvent facilement être appliqués aux autres tailles de bloc, mais ces modes exigeront leurs propres spécifications.

1.1 Conventions utilisées dans ce document

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

2. Spécification du mode CCM

Pour le mode CCM générique, il y a deux choix de paramètres. Le premier choix est M, la taille du champ d'authentification. Le choix de la valeur de M implique un compromis entre l'expansion du message et la probabilité qu'un attaquant puisse modifier un message sans être détecté. Les valeurs valides sont 4, 6, 8, 10, 12, 14, et 16 octets. Le second choix est L, la taille du champ de longueur. Cette valeur exige un compromis entre la taille maximum de message et la taille du nom occasionnel *(nonce)*. Des applications différentes auront des compromis différents, de sorte que L est un paramètre. Les valeurs valides de L vont de 2 octets à 8 octets (la valeur L=1 est réservée).

Nom	Description	Taille	Codage
M	Nombre d'octets dans le champ d'authentification	3 bits	(M-2)/2
L	Nombre d'octets dans le champ Longueur	3 bits	L-1

2.1 Entrées

Pour authentifier et chiffrer un message, les informations suivantes sont nécessaires :

1. Une clé de chiffrement K convenable pour le chiffrement de bloc.

2. Un nom occasionnel N de 15 L octets. Dans la portée de toute clé de chiffrement K, la valeur du nom occasionnel DOIT être unique. C'est-à-dire que l'ensemble des valeurs de noms occasionnels utilisées avec une certaine clé NE DOIT PAS contenir de valeur dupliquée. Utiliser le même nom occasionnel pour deux messages différents chiffrés avec la même clé détruit les propriétés de sécurité de ce mode.

3. Le message m, consistant en une chaîne de l(m) octets où $0 \leq l(m) < 2^{(8L)}$. La restriction de longueur assure que l(m) peut être codé dans un champ de L octets.

4. Les données authentifiées supplémentaires a, consistant en une chaîne de l(a) octets où $0 \leq l(a) < 2^{64}$. Ces données supplémentaires sont authentifiées mais pas chiffrées, et ne sont pas incluses dans le résultat de ce mode. Elles peuvent être utilisées pour authentifier les en-têtes de paquet de texte source, ou des informations de contexte qui affectent l'interprétation du message. Les utilisateurs qui ne souhaitent pas authentifier des données supplémentaires peuvent fournir une chaîne de longueur zéro.

Les entrées sont :

Nom	Description	Taille
K	Clé de chiffrement de bloc	Dépend du chiffrement de bloc
N	Nom occasionnel	15-L octets
m	Message à authentifier et chiffrer	l(m) octets
a	Données authentifiées supplémentaires	l(a) octets

2.2 Authentification

La première étape est de calculer le champ d'authentification T. Cela se fait en utilisant CBC-MAC [MAC]. On définit d'abord une séquence de blocs B_0, B_1, ..., B_n et ensuite on applique CBC-MAC à ces blocs.

Le premier bloc B_0 est formaté comme suit, où l(m) est codé dans l'ordre de l'octet de poids fort en premier :

Numéro d'octet	Contenu
0	Fanions
1 ... 15-L	Nom occasionnel N
16-L ... 15	l(m)

Au sein du premier bloc B_0, le champ Fanions est formaté comme suit :

Numéro de bit	Contenu
7	Réservé (toujours zéro)
6	Adata
5 ... 3	M'
2 ... 0	L'

Une autre façon de dire la même chose est : Fanions = 64*Adata + 8*M' + L'.

Le bit Réservé est réservé pour de futures expansions et devrait toujours être réglé à zéro. Le bit Adata est réglé à zéro si l(a) = 0, et à un si l(a) > 0. Le champ M' est réglé à (M-2)/2. Comme M peut prendre les valeurs paires de 4 à 16, le champ de 3 bits M' peut prendre les valeurs de un à sept. Le champ de 3 bits NE DOIT PAS avoir une valeur de zéro, qui correspondrait à une valeur de vérification d'intégrité de 16 bits. Le champ L' code la taille du champ Longueur utilisé pour mémoriser l(m). Le paramètre L peut prendre les valeurs de 2 à 8 (on rappelle que la valeur L=1 est réservée). Cette valeur est codée dans le champ de 3 bits L' en utilisant les valeurs de un à sept en choisissant L' = L-1 (la valeur zéro est réservée).

Si l(a)>0 (comme indiqué par le champ Adata) un ou plusieurs blocs de données d'authentification sont alors ajoutés. Ces blocs contiennent l(a) et sont codés de manière réversible. On construit d'abord une chaîne qui code l(a).

Si 0 < l(a) < (2^16 - 2^8), alors le champ Longueur est codé sur deux octets qui contiennent la valeur l(a) dans l'ordre de l'octet de poids fort en premier.

Si (2^16 - 2^8) ≤ l(a) < 2^32, le champ Longueur est alors codé sur six octets consistant en les octets 0xff, 0xfe, et quatre octets codants l(a) dans l'ordre de l'octet de poids fort en premier.

Si 2^32 ≤ l(a) < 2^64, alors le champ Longueur est codé sur dix octets consistant en les octets 0xff, 0xff, et huit octets qui codent l(a) dans l'ordre de l'octet de poids fort en premier.

Les conventions de codage de longueur sont résumées dans le tableau qui suit. Noter que tous les champs sont interprétés dans l'ordre de l'octet de poids fort en premier.

Deux premiers octets	Suivis par	Commentaire
0x0000	Rien	Réservé
0x0001 ... 0xFEFF	Rien	Pour 0 < l(a) < (2^16 - 2^8)
0xFF00 ... 0xFFFD	Rien	Réservé
0xFFFE	4 octets de l(a)	Pour (2^16 - 2^8) ≤ l(a) < 2^32
0xFFFF	8 octets de l(a)	Pour 2^32 ≤ l(a) < 2^64

Le codage a des bloc est formé par l'enchaînement de cette chaîne qui code l(a) avec a lui-même, et en partageant le résultat en blocs de 16 octets, puis en bourrant le dernier bloc avec des zéros si nécessaire. Ces blocs sont ajoutés au premier bloc B0.

Après que les blocs d'authentification supplémentaires (facultatifs) ont été ajoutés, on ajoute les blocs du message. Les blocs du message sont formés en partageant le message m en blocs de 16 octets, puis en bourrant le dernier bloc avec des zéros si nécessaire. Si le message m consiste en la chaîne vide, aucun bloc n'est alors ajouté à cette étape.

Le résultat est une séquence de blocs B0, B1, ..., Bn. Le CBC-MAC est calculé par :

X_1 : = E(K, B_0)
X_i+1 : = E(K, X_i OUX B_i) pour i = 1, ..., n
T : = M premiers octets de (X_n+1)

où E() est la fonction de chiffrement de chiffrement de bloc, et T est la valeur de MAC. CCM a été conçu en pensant à AES pour la fonction E(), mais tout chiffrement de bloc de 128 bits peut être utilisé. Noter que le dernier bloc B_n est OUXé avec X_n, et le résultat est chiffré avec le chiffrement de bloc. Si nécessaire, le texte chiffré est tronqué pour donner T.

2.3 Chiffrement

Pour chiffrer les données du message, on utilise le mode compteur (CTR). On définit d'abord les blocs de flux de clés par S_i : = E(K, A_i) pour i = 0, 1, 2, ...

Les valeurs A_i sont formatées comme suit, où le champ Compteur i est codé dans l'ordre de l'octet de poids fort en premier :

Numéro d'octet	Contenu
0	Fanions
1 ... 15-L	Nom occasionnel N
16-L ... 15	Compteur i

Le champ Fanions est formaté comme suit :

Numéro de bit	Contenu
7	Réservé (toujours zéro)
6	Réservé (toujours zéro)
5 ... 3	Zéro
2 ... 0	L'

Une autre façon de dire la même chose est : Fanions = L'.

Les bits réservés le sont pour de futures expansions et DOIVENT être réglés à zéro. Le bit 6 correspond au bit Adata dans le bloc B_0, mais comme ce bit n'est pas utilisé ici, il est réservé et DOIT être réglé à zéro. Les bits 3, 4, et 5 sont aussi réglés à zéro, assurant que tous les blocs A sont distincts de B_0, qui a le codage non zéro de M dans cette position. Les bits 0, 1, et 2 contiennent L', utilisant le même codage que dans B_0.

Le message est chiffré en OUixant les octets du message m avec les l(m) premiers octets de l'enchaînement de S_1, S_2, S_3, ... Noter que S_0 n'est pas utilisé pour chiffrer le message.

La valeur d'authentification U est calculée en chiffrant T avec le bloc de flux de clé S_0 et en le tronquant à la longueur désirée.

U := T OUX M premiers octets (S_0)

2.4 Résultat

Le résultat final c consiste en le message chiffré suivi par la valeur d'authentification chiffrée U.

2.5 Déchiffrement et vérification d'authentification

Pour déchiffrer un message, les informations suivantes sont nécessaires :
1. La clé de chiffrement K.
2. Le nom occasionnel N.
3. Les données authentifiées supplémentaires a.
4. Le message chiffré et authentifié c.

Le déchiffrement commence par le calcul du flux de clé pour récupérer le message m et la valeur de MAC T. Le message et les données d'authentification supplémentaires sont alors utilisées pour calculer la valeur de CBC-MAC et vérifier T.

Si la valeur T n'est pas correcte, le receveur NE DOIT PAS révéler d'informations, à part le fait que T est incorrect. Le receveur NE DOIT PAS révéler le message déchiffré, la valeur T, ou toute autre information.

2.6 Restrictions

Pour préserver la sécurité, les mises en œuvre doivent limiter la quantité totale de données chiffrées avec une seule clé ; le nombre total d'opérations de chiffrement de chiffrement de bloc dans le CBC-MAC et le chiffrement ensemble ne peuvent pas excéder 2^61. (Cela permet presque de chiffrer et authentifier 2^64 octets en utilisant CCM. C'est en gros 16 millions de tera octets, ce qui devrait être plus que suffisant pour la plupart des applications.) Dans un environnement où cette limite pourrait être atteinte, l'envoyeur DOIT s'assurer que le nombre total d'opérations de chiffrement de chiffrement de blocs dans le CBC-MAC et le chiffrement ensemble n'excèdent pas 2^61. Les receveurs qui ne s’attendent pas à déchiffrer deux fois le même message PEUVENT aussi vérifier cette limite.

Le receveur DOIT vérifier le CBC-MAC avant de révéler aucune information comme texte en clair.Si la vérification du CBC-MAC échoue, le receveur DOIT détruire toutes les informations, sauf le fait que la vérification du CBC-MAC a échoué.

3. Preuve de sécurité

Jakob Jonsson a développé une preuve de sécurité de [PROOF]. L'article résultant a été présenté à la conférence 2002 de SAC. La preuve montre que CCM fournit un niveau de confidentialité et d'authenticité qui est en ligne avec les autres modes de chiffrement authentifié proposés, comme le mode OCB [OCB].

4. Raison

La principale difficulté de la spécification de ce mode est le compromis entre la taille du nom occasionnel et la taille du compteur. Pour un mode général, on veut prendre en charge de grands messages. Certaines applications utilisent seulement de petits messages, mais auraient aussi bien un nom occasionnel plus grand. L'introduction du paramètre L résout ce problème. Le paramètre M traite le compromis traditionnel entre expansion de message et probabilité de falsification. Pour la plupart des applications, on recommande de choisir un M d'au moins 8.

Le CBC-MAC est calculé sur une séquence de blocs qui code les données pertinentes d'une façon unique. Connaissant la séquence de blocs, il est facile de récupérer N, M, L, m, et a. Le codage de la longueur de a a été choisi pour être simple et efficace quand a est vide et quand a est petit. On s'attend à ce que de nombreuses mises en œuvre limitent la taille maximum de a.

Le chiffrement CCM est une application directe du mode CTR [MODES]. Comme certaines mises en œuvre vont prendre en charge un champ de compteur de longueur variable, nous nous sommes assurés que l'octet de moindre poids du compteur est à une extrémité du champ. Cela assure aussi que le compteur est aligné sur la limite de bloc.

En chiffrant T, on évite les attaques de collision de CBC-MAC. Si le chiffrement de bloc se comporte comme une permutation pseudo aléatoire, le flux de clés est alors indistinguable d'une chaîne aléatoire. Donc, l'attaquant n'a pas d'information sur le résultat du CBC-MAC. La seule voie d'attaque qui reste est une attaque de style différentiel, qui n'a pas de chances significatives de succès si le chiffrement de bloc est une permutation pseudo aléatoire.

Pour simplifier la mise en œuvre, on utilise la même clé de chiffrement de bloc pour les fonctions de chiffrement et d'authentification. Dans notre conception, ce n'est pas un problème. Tous les blocs A sont différents, et ils sont différents du bloc B_0. Si le chiffrement de bloc se comporte comme une permutation aléatoire, les résultats sont alors indépendants les uns des autres, jusqu'à la limitation insignifiante qu'ils sont tous différents. Les seuls cas où les entrées au chiffrement de bloc peuvent se chevaucher sont sur une valeur intermédiaire dans le CBC-MAC et une provenant des autres chiffrements. Comme toutes les valeurs intermédiaires du calcul du CBC-MAC sont essentiellement aléatoires (parce que le chiffrement de bloc se comporte comme une permutation aléatoire) la probabilité d'une telle collision est très faible. Même si il y a une collision, ces valeurs n'affectent que T, qui est chiffré, de sorte qu'un attaquant ne peut pas déduire d'informations, ni détecter de collision.

On a veillé à ce que les blocs utilisés par la fonction d'authentification correspondent aux blocs utilisés par la fonction de chiffrement. Cela devrait simplifier les matériels de mise en œuvre, et réduire la quantité de glissements d'octets requise pour les mises en œuvre de logiciels.

5. Suggestions de noms occasionnels

La principale exigence est que, dans la portée d'une seule clé, les valeurs de nom occasionnel soient uniques pour chaque message. Une technique courante est de numéroter les messages en séquence, et d'utiliser ce numéro comme nom occasionnel. Les numéros de séquence de message sont aussi utilisés pour détecter les attaques en répétition et détecter le changement d'ordre des messages, de sorte que dans de nombreuses situations (comme IPsec ESP [ESP]) les numéros de séquence sont déjà disponibles.

Les utilisateurs de CCM, et de tous les autres modes de chiffrement par bloc, devraient être avertis des attaques de calcul préalable. Ce sont effectivement des attaques de collision sur la clé de chiffrement. Supposons que la clé K fasse

128 bits, et que la même valeur N' de nom occasionnel soit utilisée avec de nombreuses clés différentes. L'attaquant choisit un nom occasionnel N' particulier. Il choisit 2^64 clés différentes au hasard et calcule une entrée de tableau pour chaque valeur K, générant une paire de la forme (K,S_1). (Connaissant la clé et le nom occasionnel, le calcul de S_1 est facile.) Il attend ensuite que des messages soient envoyés avec le nom occasionnel N'. On va supposer que les 16 premiers octets de chaque message sont connus de sorte qu'il peut calculer S_1 pour chaque message. Il cherche dans son tableau une paire avec une valeur S_1 correspondante. Il peut s'attendre à trouver une correspondance après avoir vérifié environ 2^64 messages. Une fois la correspondance trouvée, l'autre partie de la paire correspondante est la clé en question. La charge de travail totale de l'attaquant est seulement de 2^64 étapes, plutôt que les 2^128 étapes attendues. Des attaques de calcul préalable similaires existent pour tous les modes de chiffrement de bloc.

La principale arme contre les attaques de calcul préalable est d'utiliser une plus grande clé. Utiliser une clé de 256 bits force l'attaquant à effectuer au moins 2^128 calculs préalables, ce qui est infaisable. Dans des situations où l'utilisation d'une grande clé n'est pas possible ou souhaitable (par exemple, à cause de l'impact résultant sur les performances) les utilisateurs peuvent utiliser une partie du nom occasionnel pour réduire le nombre de fois où une valeur spécifique de nom occasionnel est utilisée avec différentes clés. Si il y a de la place dans le nom occasionnel, l'envoyeur pourrait ajouter quelques octets aléatoires, et envoyer ces octets aléatoires avec le message. Cela rend plus difficile l'attaque de calcul préalable, car maintenant, l'attaquant doit pré calculer un tableau pour chacune des valeurs aléatoires possibles. Une autre solution est d'utiliser quelque chose comme l'adresse Ethernet de l'envoyeur. Noter que du fait de l'utilisation largement répandue de DHCP et des NAT, les adresses IP sont rarement uniques. Inclure l'adresse Ethernet force l'attaquant à effectuer le calcul préalable pour une adresse de source spécifique, et le tableau résultant ne pourra être utilisé pour attaquer personne d'autre. Bien que ces solutions puissent toutes fonctionner, il faut les analyser attentivement car elles n'empêchent jamais entièrement ces attaques. Lorsque possible, on recommande d'utiliser une plus grande clé, car cela résout tous les problèmes.

6. Efficacité et performances

Les performances dépendent de la vitesse de la mise en œuvre de chiffrement de bloc. Dans le matériel, pour les gros paquets, la vitesse réalisable pour CCM est en gros la même que celle du mode de chiffrement CBC.

Le chiffrement et l'authentification d'un message vide, sans aucune donnée d'authentification supplémentaire, exige deux opérations de chiffrement de bloc. Pour chaque bloc de données d'authentification supplémentaires est exigée une opération supplémentaire de chiffrement de bloc (si on inclut le codage de longueur). Chaque bloc de message exige deux opérations de chiffrement de bloc. Le pire des cas est lorsque le message et les données d'authentification supplémentaires sont tous deux d'un seul octet. Dans ce cas, CCM exige cinq opérations de chiffrement de bloc.

CCM résulte en l'expansion minimale possible de message ; les seuls bits ajoutés sont les bits d'authentification.

Les deux opérations de chiffrement et de déchiffrement CCM exigent seulement la fonction de chiffrement de chiffrement de bloc. En AES, les algorithmes de chiffrement et de déchiffrement ont des différences significatives. Donc, utiliser seulement l'opération de chiffrement peut conduire à des économies significatives de taille de code ou de taille de matériel.

Dans le matériel, CCM peut calculer le code d'authentification de message et effectuer le chiffrement en une seule passe. C'est-à-dire que la mise en œuvre n'a pas à achever le calcul du code d'authentification de message avant que le chiffrement puisse commencer.

CCM a été conçu pour être utilisé dans l'environnement du traitement de paquet. Le traitement de l'authentification exige que la longueur du message soit connue au début de l'opération, ce qui rend le traitement en une passe difficile dans certains environnements. Cependant, dans presque tous les environnements, les longueurs de message ou de paquet sont connues à l'avance.

7. Résumé des propriétés

Fonction de sécurité : chiffrement authentifié.
Propagation d'erreur : aucune.
Synchronisation : le même nom occasionnel est utilisé par l'envoyeur et le receveur.
Traitement en parallèle : le chiffrement peut être fait en parallèle, mais pas l'authentification.
Exigences du matériel de chiffrement : une clé.
Exigences de compteur/IV/nom occasionnel : le compteur et le nom occasionnel font partie du bloc compteur.

Exigences de mémoire : exige de la mémoire pour l'opération de chiffrement du chiffrement de bloc sous-jacent, du texte source, du texte chiffré (expansion pour CBC-MAC), et un compteur par paquet (un entier ; au plus de L octets).

Capacité de pré traitement : le flux de clé de chiffrement peut être pré calculé, mais pas l'authentification.

Exigences de longueur de message : le message est aligné sur l'octet, il est de longueur arbitraire, jusqu'à 2^(8*L) octets, les données authentifiées supplémentaires alignées sur l'octet font jusqu'à 2^64 octets.

Expansion du texte chiffré : 4, 6, 8, 10, 12, 14, ou 16 octets selon la taille du MAC choisie.

8. Vecteurs d'essai

Ces vecteurs d'essai utilisent AES pour le chiffrement de bloc [AES]. Dans chacun de ces vecteurs d'essai, les seize bits de moindre poids du bloc compteur sont utilisés pour le compteur de bloc, et le nom occasionnel fait 13 octets. Certains des vecteurs d'essai incluent une valeur d'authentification de huit octets, et d'autres incluent une valeur d'authentification de dix octets.

```
   ================ Vecteur paquet n° 1 ===================
Clé AES =  C0 C1 C2 C3  C4 C5 C6 C7  C8 C9 CA CB  CC CD CE CF
Nom occasionnel =  00 00 00 03  02 01 00 A0  A1 A2 A3 A4  A5
Longueur totale de paquet = 31. [Entré avec 8 octets d'en-tête de texte en clair]
          00 01 02 03  04 05 06 07  08 09 0A 0B  0C 0D 0E 0F
          10 11 12 13  14 15 16 17  18 19 1A 1B  1C 1D 1E
IV CBC entrant : 59 00 00 00  03 02 01 00  A0 A1 A2 A3  A4 A5 00 17
IV CBC sortant :EB 9D 55 47  73 09 55 AB  23 1E 0A 2D  FE 4B 90 D6
Après OUx : EB 95 55 46  71 0A 51 AE  25 19 0A 2D  FE 4B 90 D6   [en-tête]
Après AES : CD B6 41 1E  3C DC 9B 4F  5D 92 58 B6  9E E7 F0 91
Après OUx : C5 BF 4B 15  30 D1 95 40  4D 83 4A A5  8A F2 E6 86   [message]
Après AES : 9C 38 40 5E  A0 3C 1B C9  04 B5 8B 40  C7 6C A2 EB
Après OUx : 84 21 5A 45  BC 21 05 C9  04 B5 8B 40  C7 6C A2 EB   [message]
Après AES : 2D C6 97 E4  11 CA 83 A8  60 C2 C4 06  CC AA 54 2F
CBC-MAC : 2D C6 97 E4  11 CA 83 A8
Début CTR : 01 00 00 00  03 02 01 00  A0 A1 A2 A3  A4 A5 00 01
CTR[0001]: 50 85 9D 91  6D CB 6D DD  E0 77 C2 D1  D4 EC 9F 97
CTR[0002]: 75 46 71 7A  C6 DE 9A FF  64 0C 9C 06  DE 6D 0D 8F
CTR[MAC ]: 3A 2E 46 C8  EC 33 A5 48
Longueur totale de paquet = 39. [Résultat authentifié et chiffré]
          00 01 02 03  04 05 06 07  58 8C 97 9A  61 C6 63 D2
          F0 66 D0 C2  C0 F9 89 80  6D 5F 6B 61  DA C3 84 17
          E8 D1 2C FD  F9 26 E0

   ================ Vecteur paquet n° 2 ===================
Clé AES =  C0 C1 C2 C3  C4 C5 C6 C7  C8 C9 CA CB  CC CD CE CF
Nom occasionnel = 00 00 00 04  03 02 01 A0  A1 A2 A3 A4  A5
Longueur totale de paquet = 32. [Entré avec 8 octets d'en-tête de texte en clair]
          00 01 02 03  04 05 06 07  08 09 0A 0B  0C 0D 0E 0F
          10 11 12 13  14 15 16 17  18 19 1A 1B  1C 1D 1E 1F
IV CBC entrant : 59 00 00 00  04 03 02 01  A0 A1 A2 A3  A4 A5 00 18
IV CBC sortant :F0 C2 54 D3  CA 03 E2 39  70 BD 24 A8  4C 39 9E 77
Après OUx : F0 CA 54 D2  C8 00 E6 3C  76 BA 24 A8  4C 39 9E 77   [en-tête]
Après AES : 48 DE 8B 86  28 EA 4A 40  00 AA 42 C2  95 BF 4A 8C
Après OUx : 40 D7 81 8D  24 E7 44 4F  10 BB 50 D1  81 AA 5C 9B   [message]
Après AES : 0F 89 FF BC  A6 2B C2 4F  13 21 5F 16  87 96 AA 33
Après OUx : 17 90 E5 A7  BA 36 DC 50  13 21 5F 16  87 96 AA 33   [message]
Après AES : F7 B9 05 6A  86 92 6C F3  FB 16 3D C4  99 EF AA 11
CBC-MAC : F7 B9 05 6A  86 92 6C F3
Début CTR : 01 00 00 00  04 03 02 01  A0 A1 A2 A3  A4 A5 00 01
CTR[0001] : 7A C0 10 3D  ED 38 F6 C0  39 0D BA 87  1C 49 91 F4
CTR[0002] : D4 0C DE 22  D5 F9 24 24  F7 BE 9A 56  9D A7 9F 51
CTR[MAC] : 57 28 D0 04  96 D2 65 E5
Longueur totale de paquet = 40. [résultat authentifié et chiffré]
          00 01 02 03  04 05 06 07  72 C9 1A 36  E1 35 F8 CF
```

```
        29 1C A8 94  08 5C 87 E3  CC 15 C4 39  C9 E4 3A 3B
        A0 91 D5 6E  10 40 09 16

  ================ Vecteur paquet n° 3 ===================
Clé AES =  C0 C1 C2 C3  C4 C5 C6 C7  C8 C9 CA CB  CC CD CE CF
Nom occasionnel =  00 00 00 05  04 03 02 A0  A1 A2 A3 A4  A5
Longueur totale de paquet = 33. [Entré avec 8 octets d'en-tête de texte en clair]
        00 01 02 03  04 05 06 07  08 09 0A 0B  0C 0D 0E 0F
        10 11 12 13  14 15 16 17  18 19 1A 1B  1C 1D 1E 1F
        20
IV CBC entrant : 59 00 00 00  05 04 03 02  A0 A1 A2 A3  A4 A5 00 19
IV CBC sortant :6F 8A 12 F7  BF 8D 4D C5  A1 19 6E 95  DF F0 B4 27
Après OUx : 6F 82 12 F6  BD 8E 49 C0  A7 1E 6E 95  DF F0 B4 27   [en-tête]
Après AES : 37 E9 B7 8C  C2 20 17 E7  33 80 43 0C  BE F4 28 24
Après OUx : 3F E0 BD 87  CE 2D 19 E8  23 91 51 1F  AA E1 3E 33   [message]
Après AES : 90 CA 05 13  9F 4D 4E CF  22 6F E9 81  C5 9E 2D 40
Après OUx : 88 D3 1F 08  83 50 50 D0  02 6F E9 81  C5 9E 2D 40   [message]
Après AES : 73 B4 67 75  C0 26 DE AA  41 03 97 D6  70 FE 5F B0
CBC-MAC : 73 B4 67 75  C0 26 DE AA
Début CTR : 01 00 00 00  05 04 03 02  A0 A1 A2 A3  A4 A5 00 01
CTR[0001] : 59 B8 EF FF  46 14 73 12  B4 7A 1D 9D  39 3D 3C FF
CTR[0002] : 69 F1 22 A0  78 C7 9B 89  77 89 4C 99  97 5C 23 78
CTR[MAC] : 39 6E C0 1A  7D B9 6E 6F
Longueur totale de paquet = 41. [résultat authentifié et chiffré]
        00 01 02 03  04 05 06 07  51 B1 E5 F4  4A 19 7D 1D
        A4 6B 0F 8E  2D 28 2A E8  71 E8 38 BB  64 DA 85 96
        57 4A DA A7  6F BD 9F B0  C5

  ================ Vecteur paquet n° 4 ===================
Clé AES =  C0 C1 C2 C3  C4 C5 C6 C7  C8 C9 CA CB  CC CD CE CF
Nom occasionnel = 00 00 00 06  05 04 03 A0  A1 A2 A3 A4  A5
Longueur totale de paquet = 31. [Entré avec 12 octets d'en-tête de texte en clair]
        00 01 02 03  04 05 06 07  08 09 0A 0B  0C 0D 0E 0F
        10 11 12 13  14 15 16 17  18 19 1A 1B  1C 1D 1E
IV CBC entrant : 59 00 00 00  06 05 04 03  A0 A1 A2 A3  A4 A5 00 13
IV CBC sortant :06 65 2C 60  0E F5 89 63  CA C3 25 A9  CD 3E 2B E1
Après OUx : 06 69 2C 61  0C F6 8D 66  CC C4 2D A0  C7 35 2B E1   [en-tête]
Après AES : A0 75 09 AC  15 C2 58 86  04 2F 80 60  54 FE A6 86
Après OUx : AC 78 07 A3  05 D3 4A 95  10 3A 96 77  4C E7 BC 9D   [message]
Après AES : 64 4C 09 90  D9 1B 83 E9  AB 4B 8E ED  06 6F F5 BF
Après OUx : 78 51 17 90  D9 1B 83 E9  AB 4B 8E ED  06 6F F5 BF   [message]
Après AES : 4B 4F 4B 39  B5 93 E6 BF  B0 B2 C2 B7  0F 29 CD 7A
CBC-MAC : 4B 4F 4B 39  B5 93 E6 BF
Début CTR : 01 00 00 00  06 05 04 03  A0 A1 A2 A3  A4 A5 00 01
CTR[0001] : AE 81 66 6A  83 8B 88 6A  EE BF 4A 5B  32 84 50 8A
CTR[0002] : D1 B1 92 06  AC 93 9E 2F  B6 DD CE 10  A7 74 FD 8D
CTR[MAC] : DD 87 2A 80  7C 75 F8 4E
Longueur totale de paquet = 39. [résultat authentifié et chiffré]
        00 01 02 03  04 05 06 07  08 09 0A 0B  A2 8C 68 65
        93 9A 9A 79  FA AA 5C 4C  2A 9D 4A 91  CD AC 8C 96
        C8 61 B9 C9  E6 1E F1

  ================ Vecteur paquet n° 5 ===================
Clé AES =  C0 C1 C2 C3  C4 C5 C6 C7  C8 C9 CA CB  CC CD CE CF
Nom occasionnel = 00 00 00 07  06 05 04 A0  A1 A2 A3 A4  A5
Longueur totale de paquet = 32. [Entré avec 12 octets d'en-tête de texte en clair]
        00 01 02 03  04 05 06 07  08 09 0A 0B  0C 0D 0E 0F
        10 11 12 13  14 15 16 17  18 19 1A 1B  1C 1D 1E 1F
IV CBC entrant : 59 00 00 00  07 06 05 04  A0 A1 A2 A3  A4 A5 00 14
IV CBC sortant :00 4C 50 95  45 80 3C 48  51 CD E1 3B  56 C8 9A 85
Après OUx : 00 40 50 94  47 83 38 4D  57 CA E9 32  5C C3 9A 85   [en-tête]
Après AES : E2 B8 F7 CE  49 B2 21 72  84 A8 EA 84  FA AD 67 5C
```

Après OUx : EE B5 F9 C1 59 A3 33 61 90 BD FC 93 E2 B4 7D 47 [message]
Après AES : 3E FB 36 72 25 DB 11 01 D3 C2 2F 0E CA FF 44 F3
Après OUx : 22 E6 28 6D 25 DB 11 01 D3 C2 2F 0E CA FF 44 F3 [message]
Après AES : 48 B9 E8 82 55 05 4A B5 49 0A 95 F9 34 9B 4B 5E
CBC-MAC : 48 B9 E8 82 55 05 4A B5
Début CTR : 01 00 00 00 07 06 05 04 A0 A1 A2 A3 A4 A5 00 01
CTR[0001] : D0 FC F5 74 4D 8F 31 E8 89 5B 05 05 4B 7C 90 C3
CTR[0002] : 72 A0 D4 21 9F 0D E1 D4 04 83 BC 2D 3D 0C FC 2A
CTR[MAC] : 19 51 D7 85 28 99 67 26
Longueur totale de paquet = 40. [résultat authentifié et chiffré]
00 01 02 03 04 05 06 07 08 09 0A 0B DC F1 FB 7B
5D 9E 23 FB 9D 4E 13 12 53 65 8A D8 6E BD CA 3E
51 E8 3F 07 7D 9C 2D 93

========== Vecteur paquet n° 6 ==========
Clé AES = C0 C1 C2 C3 C4 C5 C6 C7 C8 C9 CA CB CC CD CE CF
Nom occasionnel = 00 00 00 08 07 06 05 A0 A1 A2 A3 A4 A5
Longueur totale de paquet = 33. [Entré avec 12 octets d'en-tête de texte en clair]
00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F
20
IV CBC entrant : 59 00 00 00 08 07 06 05 A0 A1 A2 A3 A4 A5 00 15
IV CBC sortant :04 72 DA 4C 6F F6 0A 63 06 52 1A 06 04 80 CD E5
Après OUx : 04 7E DA 4D 6D F5 0E 66 00 55 12 0F 0E 8B CD E5 [en-tête]
Après AES : 64 4C 36 A5 A2 27 37 62 0B 89 F1 D7 BF F2 73 D4
Après OUx : 68 41 38 AA B2 36 25 71 1F 9C E7 C0 A7 EB 69 CF [message]
Après AES : 41 E1 19 CD 19 24 CE 77 F1 2F A6 60 C1 6E BB 4E
Après OUx : 5D FC 07 D2 39 24 CE 77 F1 2F A6 60 C1 6E BB 4E [message]
Après AES : A5 27 D8 15 6A C3 59 BF 1C B8 86 E6 2F 29 91 29
CBC-MAC : A5 27 D8 15 6A C3 59 BF
Début CTR : 01 00 00 00 08 07 06 05 A0 A1 A2 A3 A4 A5 00 01
CTR[0001] : 63 CC BE 1E E0 17 44 98 45 64 B2 3A 8D 24 5C 80
CTR[0002] : 39 6D BA A2 A7 D2 CB D4 B5 E1 7C 10 79 45 BB C0
CTR[MAC] : E5 7D DC 56 C6 52 92 2B
Longueur totale de paquet = 41. [résultat authentifié et chiffré]
00 01 02 03 04 05 06 07 08 09 0A 0B 6F C1 B0 11
F0 06 56 8B 51 71 A4 2D 95 3D 46 9B 25 70 A4 BD
87 40 5A 04 43 AC 91 CB 94

========== Vecteur paquet n° 7 ==========
Clé AES = C0 C1 C2 C3 C4 C5 C6 C7 C8 C9 CA CB CC CD CE CF
Nom occasionnel = 00 00 00 09 08 07 06 A0 A1 A2 A3 A4 A5
Longueur totale de paquet = 31. [Entré avec 8 octets d'en-tête de texte en clair]
00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E
IV CBC entrant : 61 00 00 00 09 08 07 06 A0 A1 A2 A3 A4 A5 00 17
IV CBC sortant :60 06 C5 72 DA 23 9C BF A0 5B 0A DE D2 CD A8 1E
Après OUx : 60 0E C5 73 D8 20 98 BA A6 5C 0A DE D2 CD A8 1E [en-tête]
Après AES : 41 7D E2 AE 94 E2 EA D9 00 FC 44 FC D0 69 52 27
Après OUx : 49 74 E8 A5 98 EF E4 D6 10 ED 56 EF C4 7C 44 30 [message]
Après AES : 2A 6C 42 CA 49 D7 C7 01 C5 7D 59 FF 87 16 49 0E
Après OUx : 32 75 58 D1 55 CA D9 01 C5 7D 59 FF 87 16 49 0E [message]
Après AES : 89 8B D6 45 4E 27 20 BB D2 7E F3 15 7A 7C 90 B2
CBC-MAC : 89 8B D6 45 4E 27 20 BB D2 7E
Début CTR : 01 00 00 00 09 08 07 06 A0 A1 A2 A3 A4 A5 00 01
CTR[0001] : 09 3C DB B9 C5 52 4F DA C1 C5 EC D2 91 C4 70 AF
CTR[0002] : 11 57 83 86 E2 C4 72 B4 8E CC 8A AD AB 77 6F CB
CTR[MAC] : 8D 07 80 25 62 B0 8C 00 A6 EE
Longueur totale de paquet = 41. [résultat authentifié et chiffré]
00 01 02 03 04 05 06 07 01 35 D1 B2 C9 5F 41 D5
D1 D4 FE C1 85 D1 66 B8 09 4E 99 9D FE D9 6C 04
8C 56 60 2C 97 AC BB 74 90

```
   ================ Vecteur paquet n° 8 ==================
Clé AES =  C0 C1 C2 C3  C4 C5 C6 C7  C8 C9 CA CB  CC CD CE CF
Nom occasionnel = 00 00 00 0A  09 08 07 A0  A1 A2 A3 A4  A5
Longueur totale de paquet = 32. [Entré avec 8 octets d'en-tête de texte en clair]
          00 01 02 03  04 05 06 07  08 09 0A 0B  0C 0D 0E 0F
          10 11 12 13  14 15 16 17  18 19 1A 1B  1C 1D 1E 1F
IV CBC entrant : 61 00 00 00  0A 09 08 07  A0 A1 A2 A3  A4 A5 00 18
IV CBC sortant :63 A3 FA E4  6C 79 F3 FA  78 38 B8 A2  80 36 B6 0B
Après OUx : 63 AB FA E5  6E 7A F7 FF  7E 3F B8 A2  80 36 B6 0B   [en-tête]
Après AES : 1C 99 1A 3D  B7 60 79 27  34 40 79 1F  AD 8B 5B 02
Après OUx : 14 90 10 36  BB 6D 77 28  24 51 6B 0C  B9 9E 4D 15   [message]
Après AES : 14 19 E8 E8  CB BE 75 58  E1 E3 BE 4B  6C 9F 82 E3
Après OUx : 0C 00 F2 F3  D7 A3 6B 47  E1 E3 BE 4B  6C 9F 82 E3   [message]
Après AES : E0 16 E8 1C  7F 7B 8A 38  A5 38 F2 CB  5B B6 C1 F2
CBC-MAC : E0 16 E8 1C  7F 7B 8A 38  A5 38
Début CTR : 01 00 00 00  0A 09 08 07  A0 A1 A2 A3  A4 A5 00 01
CTR[0001] : 73 7C 33 91  CC 8E 13 DD  E0 AA C5 4B  6D B7 EB 98
CTR[0002] : 74 B7 71 77  C5 AA C5 3B  04 A4 F8 70  8E 92 EB 2B
CTR[MAC] : 21 6D AC 2F  8B 4F 1C 07  91 8C
Longueur totale de paquet = 42. [résultat authentifié et chiffré]
          00 01 02 03  04 05 06 07  7B 75 39 9A  C0 83 1D D2
          F0 BB D7 58  79 A2 FD 8F  6C AE 6B 6C  D9 B7 DB 24
          C1 7B 44 33  F4 34 96 3F  34 B4

   ================ Vecteur paquet n° 9 ==================
Clé AES =  C0 C1 C2 C3  C4 C5 C6 C7  C8 C9 CA CB  CC CD CE CF
Nom occasionnel = 00 00 00 0B  0A 09 08 A0  A1 A2 A3 A4  A5
Longueur totale de paquet = 33. [Entré avec 8 octets d'en-tête de texte en clair]
          00 01 02 03  04 05 06 07  08 09 0A 0B  0C 0D 0E 0F
          10 11 12 13  14 15 16 17  18 19 1A 1B  1C 1D 1E 1F
          20
IV CBC entrant : 61 00 00 00  0B 0A 09 08  A0 A1 A2 A3  A4 A5 00 19
IV CBC sortant :4F 2C 86 11  1E 08 2A DD  6B 44 21 3A  B5 13 13 16
Après OUx : 4F 24 86 10  1C 0B 2E D8  6D 43 21 3A  B5 13 13 16   [en-tête]
Après AES : F6 EC 56 87  3C 57 12 DC  9C C5 3C A8  D4 D1 ED 0A
Après OUx : FE E5 5C 8C  30 5A 1C D3  8C D4 2E BB  C0 C4 FB 1D   [message]
Après AES : 17 C1 80 A5  31 53 D4 C3  03 85 0C 95  65 80 34 52
Après OUx : 0F D8 9A BE  2D 4E CA DC  23 85 0C 95  65 80 34 52   [message]
Après AES : 46 A1 F6 E2  B1 6E 75 F8  1C F5 6B 1A  80 04 44 1B
CBC-MAC : 46 A1 F6 E2  B1 6E 75 F8  1C F5
Début CTR : 01 00 00 00  0B 0A 09 08  A0 A1 A2 A3  A4 A5 00 01
CTR[0001] : 8A 5A 10 6B  C0 29 9A 55  5B 93 6B 0B  0E A0 DE 5A
CTR[0002] : EA 05 FD E2  AB 22 5C FE  B7 73 12 CB  88 D9 A5 4A
CTR[MAC] : AC 3D F1 07  DA 30 C4 86  43 BB
Longueur totale de paquet = 43. [résultat authentifié et chiffré]
          00 01 02 03  04 05 06 07  82 53 1A 60  CC 24 94 5A
          4B 82 79 18  1A B5 C8 4D  F2 1C E7 F9  B7 3F 42 E1
          97 EA 9C 07  E5 6B 5E B1  7E 5F 4E

   ================ Vecteur paquet n° 10 =================
Clé AES =  C0 C1 C2 C3  C4 C5 C6 C7  C8 C9 CA CB  CC CD CE CF
Nom occasionnel = 00 00 00 0C  0B 0A 09 A0  A1 A2 A3 A4  A5
Longueur totale de paquet = 31. [Entré avec 12 octets d'en-tête de texte en clair]
          00 01 02 03  04 05 06 07  08 09 0A 0B  0C 0D 0E 0F
          10 11 12 13  14 15 16 17  18 19 1A 1B  1C 1D 1E
IV CBC entrant : 61 00 00 00  0C 0B 0A 09  A0 A1 A2 A3  A4 A5 00 13
IV CBC sortant :7F B8 0A 32  E9 80 57 46  EC 31 6C 3A  B2 A2 EB 5D
Après OUx : 7F B4 0A 33  EB 83 53 43  EA 36 64 33  B8 A9 EB 5D   [en-tête]
Après AES : 7E 96 96 BF  F1 56 D6 A8  6E AC F5 7B  7F 23 47 5A
Après OUx : 72 9B 98 B0  E1 47 C4 BB  7A B9 E3 6C  67 3A 5D 41   [message]
Après AES : 8B 4A EE 42  04 24 8A 59  FA CC 88 66  57 66 DD 72
```

Après OUx : 97 57 F0 42 04 24 8A 59 FA CC 88 66 57 66 DD 72 [message]
Après AES : 41 63 89 36 62 ED D7 EB CD 6E 15 C1 89 48 62 05
CBC-MAC : 41 63 89 36 62 ED D7 EB CD 6E
Début CTR : 01 00 00 00 0C 0B 0A 09 A0 A1 A2 A3 A4 A5 00 01
CTR[0001] : 0B 39 2B 9B 05 66 97 06 3F 12 56 8F 2B 13 A1 0F
CTR[0002] : 07 89 65 25 23 40 94 3B 9E 69 B2 56 CC 5E F7 31
CTR[MAC] : 17 09 20 76 09 A0 4E 72 45 B3
Longueur totale de paquet = 41. [résultat authentifié et chiffré]
00 01 02 03 04 05 06 07 08 09 0A 0B 07 34 25 94
15 77 85 15 2B 07 40 98 33 0A BB 14 1B 94 7B 56
6A A9 40 6B 4D 99 99 88 DD

========== Vecteur paquet n° 11 ==========
Clé AES = C0 C1 C2 C3 C4 C5 C6 C7 C8 C9 CA CB CC CD CE CF
Nom occasionnel = 00 00 00 0D 0C 0B 0A A0 A1 A2 A3 A4 A5
Longueur totale de paquet = 32. [Entré avec 12 octets d'en-tête de texte en clair]
00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F
IV CBC entrant : 61 00 00 00 0D 0C 0B 0A A0 A1 A2 A3 A4 A5 00 14
IV CBC sortant :B0 84 85 79 51 D2 FA 42 76 EF 3A D7 14 B9 62 87
Après OUx : B0 88 85 78 53 D1 FE 47 70 E8 32 DE 1E B2 62 87 [en-tête]
Après AES : C9 B3 64 7E D8 79 2A 5C 65 B7 CE CC 19 0A 97 0A
Après OUx : C5 BE 6A 71 C8 68 38 4F 71 A2 D8 DB 01 13 8D 11 [message]
Après AES : 34 0F 69 17 FA B9 19 D6 1D AC D0 35 36 D6 55 8B
Après OUx : 28 12 77 08 FA B9 19 D6 1D AC D0 35 36 D6 55 8B [message]
Après AES : 6B 5E 24 34 12 CC C2 AD 6F 1B 11 C3 A1 A9 D8 BC
CBC-MAC : 6B 5E 24 34 12 CC C2 AD 6F 1B
Début CTR : 01 00 00 00 0D 0C 0B 0A A0 A1 A2 A3 A4 A5 00 01
CTR[0001] : 6B 66 BC 0C 90 A1 F1 12 FC BE 6F 4E 12 20 77 BC
CTR[0002] : 97 9E 57 2B BE 65 8A E5 CC 20 11 83 2A 9A 9B 5B
CTR[MAC] : 9E 64 86 DD 02 B6 49 C1 6D 37
Longueur totale de paquet = 42. [résultat authentifié et chiffré]
00 01 02 03 04 05 06 07 08 09 0A 0B 67 6B B2 03
80 B0 E3 01 E8 AB 79 59 0A 39 6D A7 8B 83 49 34
F5 3A A2 E9 10 7A 8B 6C 02 2C

========== Vecteur paquet n° 12 ==========
Clé AES = C0 C1 C2 C3 C4 C5 C6 C7 C8 C9 CA CB CC CD CE CF
Nom occasionnel = 00 00 00 0E 0D 0C 0B A0 A1 A2 A3 A4 A5
Longueur totale de paquet = 33. [Entré avec 12 octets d'en-tête de texte en clair]
00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F
20
IV CBC entrant : 61 00 00 00 0E 0D 0C 0B A0 A1 A2 A3 A4 A5 00 15
IV CBC sortant :5F 8E 8D 02 AD 95 7C 5A 36 14 CF 63 40 16 97 4F
Après OUx : 5F 82 8D 03 AF 96 78 5F 30 13 C7 6A 4A 1D 97 4F [en-tête]
Après AES : 63 FA BD 69 B9 55 65 FF 54 AA F4 60 88 7D EC 9F
Après OUx : 6F F7 B3 66 A9 44 77 EC 40 BF E2 77 90 64 F6 84 [message]
Après AES : 5A 76 5F 0B 93 CE 4F 6A B4 1D 91 30 18 57 6A D7
Après OUx : 46 6B 41 14 B3 CE 4F 6A B4 1D 91 30 18 57 6A D7 [message]
Après AES : 9D 66 92 41 01 08 D5 B6 A1 45 85 AC AF 86 32 E8
CBC-MAC : 9D 66 92 41 01 08 D5 B6 A1 45
Début CTR : 01 00 00 00 0E 0D 0C 0B A0 A1 A2 A3 A4 A5 00 01
CTR[0001] : CC F2 AE D9 E0 4A C9 74 E6 58 55 B3 2B 94 30 BF
CTR[0002] : A2 CA AC 11 63 F4 07 E5 E5 F6 E3 B3 79 0F 79 F8
CTR[MAC] : 50 7C 31 57 63 EF 78 D3 77 9E
Longueur totale de paquet = 43. [résultat authentifié et chiffré]
00 01 02 03 04 05 06 07 08 09 0A 0B C0 FF A0 D6
F0 5B DB 67 F2 4D 43 A4 33 8D 2A A4 BE D7 B2 0E
43 CD 1A A3 16 62 E7 AD 65 D6 DB

```
   ================ Vecteur paquet n° 13 ===================
Clé AES =  D7 82 8D 13  B2 B0 BD C3  25 A7 62 36  DF 93 CC 6B
Nom occasionnel = 00 41 2B 4E  A9 CD BE 3C  96 96 76 6C  FA
Longueur totale de paquet = 31. [Entré avec 8 octets d'en-tête de texte en clair]
          0B E1 A8 8B  AC E0 18 B1  08 E8 CF 97  D8 20 EA 25
          84 60 E9 6A  D9 CF 52 89  05 4D 89 5C  EA C4 7C
IV CBC entrant : 59 00 41 2B  4E A9 CD BE  3C 96 96 76  6C FA 00 17
IV CBC sortant :33 AE C3 1A  1F B7 CC 35  E5 DA D2 BA  C0 90 D9 A3
Après OUx : 33 A6 C8 FB  B7 3C 60 D5  FD 6B D2 BA  C0 90 D9 A3   [en-tête]
Après AES : B7 56 CA 1E  5B 42 C6 9C  58 E3 0A F5  2B F7 7C FD
Après OUx : BF BE 05 89  83 62 2C B9  DC 83 E3 9F  F2 38 2E 74   [message]
Après AES : 33 3D 3A 3D  07 B5 3C 7B  22 0E 96 1A  18 A9 A1 9E
Après OUx : 36 70 B3 61  ED 71 40 7B  22 0E 96 1A  18 A9 A1 9E   [message]
Après AES : 14 BD DB 6B  F9 01 63 4D  FB 56 51 83  BC 74 93 F7
CBC-MAC : 14 BD DB 6B  F9 01 63 4D
Début CTR : 01 00 41 2B  4E A9 CD BE  3C 96 96 76  6C FA 00 01
CTR[0001] : 44 51 B0 11  7A 84 82 BF  03 19 AE C1  59 5E BD DA
CTR[0002] : 83 EB 76 E1  3A 44 84 7F  92 20 09 07  76 B8 25 C5
CTR[MAC] : F3 31 2C A0  F5 DC B4 FE
Longueur totale de paquet = 39. [résultat authentifié et chiffré]
          0B E1 A8 8B  AC E0 18 B1  4C B9 7F 86  A2 A4 68 9A
          87 79 47 AB  80 91 EF 53  86 A6 FF BD  D0 80 F8 E7
          8C F7 CB 0C  DD D7 B3

   ================ Vecteur paquet n° 14 ===================
Clé AES =  D7 82 8D 13  B2 B0 BD C3  25 A7 62 36  DF 93 CC 6B
Nom occasionnel = 00 33 56 8E  F7 B2 63 3C  96 96 76 6C  FA
Longueur totale de paquet = 32. [Entré avec 8 octets d'en-tête de texte en clair]
          63 01 8F 76  DC 8A 1B CB  90 20 EA 6F  91 BD D8 5A
          FA 00 39 BA  4B AF F9 BF  B7 9C 70 28  94 9C D0 EC
IV CBC entrant : 59 00 33 56  8E F7 B2 63  3C 96 96 76  6C FA 00 18
IV CBC sortant :42 0D B1 50  BB 0C 44 DA  83 E4 52 09  55 99 67 E3
Après OUx : 42 05 D2 51  34 7A 98 50  98 2F 52 09  55 99 67 E3   [en-tête]
Après AES : EA D1 CA 56  02 02 09 5C  E6 12 B0 D2  18 A0 DD 44
Après OUx : 7A F1 20 39  93 BF D1 06  1C 12 89 68  53 0F 24 FB   [message]
Après AES : 51 77 41 69  C3 DE 6B 24  13 27 74 90  F5 FF C5 62
Après OUx : E6 EB 31 41  57 42 BB C8  13 27 74 90  F5 FF C5 62   [message]
Après AES : D4 CC 3B 82  DF 9F CC 56  7E E5 83 61  D7 8D FB 5E
CBC-MAC : D4 CC 3B 82  DF 9F CC 56
Début CTR : 01 00 33 56  8E F7 B2 63  3C 96 96 76  6C FA 00 01
CTR[0001] : DC EB F4 13  38 3C 66 A0  5A 72 55 EF  98 D7 FF AD
CTR[0002] : 2F 54 2C BA  15 D6 6C DF  E1 EC 46 8F  0E 68 A1 24
CTR[MAC] : 11 E2 D3 9F  A2 E8 0C DC
Longueur totale de paquet = 40. [résultat authentifié et chiffré]
          63 01 8F 76  DC 8A 1B CB  4C CB 1E 7C  A9 81 BE FA
          A0 72 6C 55  D3 78 06 12  98 C8 5C 92  81 4A BC 33
          C5 2E E8 1D  7D 77 C0 8A

   ================ Vecteur paquet n° 15 ===================
Clé AES =  D7 82 8D 13  B2 B0 BD C3  25 A7 62 36  DF 93 CC 6B
Nom occasionnel =    00 10 3F E4  13 36 71 3C  96 96 76 6C  FA
Longueur totale de paquet = 33. [Entré avec 8 octets d'en-tête de texte en clair]
          AA 6C FA 36  CA E8 6B 40  B9 16 E0 EA  CC 1C 00 D7
          DC EC 68 EC  0B 3B BB 1A  02 DE 8A 2D  1A A3 46 13
          2E
IV CBC entrant : 59 00 10 3F  E4 13 36 71  3C 96 96 76  6C FA 00 19
IV CBC sortant :B3 26 49 FF  D5 9F 56 0F  02 2D 11 E2  62 C5 BE EA
Après OUx : B3 2E E3 93  2F A9 9C E7  69 6D 11 E2  62 C5 BE EA   [en-tête]
Après AES : 82 50 9E E5  B2 FF DB CA  9B D0 2E 20  6B 3F B7 AD
Après OUx : 3B 46 7E 0F  7E E3 DB 1D  47 3C 46 CC  60 04 0C B7   [message]
Après AES : 80 46 0E 4C  08 3A D0 3F  B9 A9 13 BE  E4 DE 2F 66
Après OUx : 82 98 84 61  12 99 96 2C  97 A9 13 BE  E4 DE 2F 66   [message]
```

Après AES : 47 29 CB 00 31 F1 81 C1 92 68 4B 89 A4 71 50 E7
CBC-MAC : 47 29 CB 00 31 F1 81 C1
Début CTR : 01 00 10 3F E4 13 36 71 3C 96 96 76 6C FA 00 01
CTR[0001] : 08 C4 DA C8 EC C1 C0 7B 4C E1 F2 4C 37 5A 47 EE
CTR[0002] : A7 87 2E 6C 6D C4 4E 84 26 02 50 4C 3F A5 73 C5
CTR[MAC] : E0 5F B2 6E EA 83 B4 C7
Longueur totale de paquet = 41. [résultat authentifié et chiffré]
AA 6C FA 36 CA E8 6B 40 B1 D2 3A 22 20 DD C0 AC
90 0D 9A A0 3C 61 FC F4 A5 59 A4 41 77 67 08 97
08 A7 76 79 6E DB 72 35 06

========== Vecteur paquet n° 16 ==========
Clé AES = D7 82 8D 13 B2 B0 BD C3 25 A7 62 36 DF 93 CC 6B
Nom occasionnel = 00 76 4C 63 B8 05 8E 3C 96 96 76 6C FA
Longueur totale de paquet = 31. [Entré avec 12 octets d'en-tête de texte en clair]
D0 D0 73 5C 53 1E 1B EC F0 49 C2 44 12 DA AC 56
30 EF A5 39 6F 77 0C E1 A6 6B 21 F7 B2 10 1C
IV CBC entrant : 59 00 76 4C 63 B8 05 8E 3C 96 96 76 6C FA 00 13
IV CBC sortant :AB DC 4E C9 AA 72 33 97 DF 2D AD 76 33 DE 3B 0D
Après OUx : AB D0 9E 19 D9 2E 60 89 C4 C1 5D 3F F1 9A 3B 0D [en-tête]
Après AES : 62 86 F6 2F 23 42 63 B0 1C FD 8C 37 40 74 81 EB
Après OUx : 70 5C 5A 79 13 AD C6 89 73 8A 80 D6 E6 1F A0 1C [message]
Après AES : 88 95 84 18 CF 79 CA BE EB C0 0C C4 86 E6 01 F7
Après OUx : 3A 85 98 18 CF 79 CA BE EB C0 0C C4 86 E6 01 F7 [message]
Après AES : C1 85 92 D9 84 CD 67 80 63 D1 D9 6D C1 DF A1 11
CBC-MAC : C1 85 92 D9 84 CD 67 80
Début CTR : 01 00 76 4C 63 B8 05 8E 3C 96 96 76 6C FA 00 01
CTR[0001] : 06 08 FF 95 A6 94 D5 59 F4 0B B7 9D EF FA 41 DF
CTR[0002] : 80 55 3A 75 78 38 04 A9 64 8B 68 DD 7F DC DD 7A
CTR[MAC] : 5B EA DB 4E DF 07 B9 2F
Longueur totale de paquet = 39. [résultat authentifié et chiffré]
D0 D0 73 5C 53 1E 1B EC F0 49 C2 44 14 D2 53 C3
96 7B 70 60 9B 7C BB 7C 49 91 60 28 32 45 26 9A
6F 49 97 5B CA DE AF

========== Vecteur paquet n° 17 ==========
Clé AES = D7 82 8D 13 B2 B0 BD C3 25 A7 62 36 DF 93 CC 6B
Nom occasionnel = 00 F8 B6 78 09 4E 3B 3C 96 96 76 6C FA
Longueur totale de paquet = 32. [Entré avec 12 octets d'en-tête de texte en clair]
77 B6 0F 01 1C 03 E1 52 58 99 BC AE E8 8B 6A 46
C7 8D 63 E5 2E B8 C5 46 EF B5 DE 6F 75 E9 CC 0D
IV CBC entrant : 59 00 F8 B6 78 09 4E 3B 3C 96 96 76 6C FA 00 14
IV CBC sortant :F4 68 FE 5D B1 53 0B 7A 5A A5 FB 27 40 CF 6E 33
Après OUx : F4 64 89 EB BE 52 17 79 BB F7 A3 BE FC 61 6E 33 [en-tête]
Après AES : 23 29 0E 0B 33 45 9A 83 32 2D E4 06 86 67 10 04
Après OUx : CB A2 64 4D F4 C8 F9 66 1C 95 21 40 69 D2 CE 6B [message]
Après AES : 8F BE D4 0F 8B 89 B7 B8 20 D5 5F E0 3C E2 43 11
Après OUx : FA 57 18 02 8B 89 B7 B8 20 D5 5F E0 3C E2 43 11 [message]
Après AES : 6A DB 15 B6 71 81 B2 E2 2B E3 4A F2 B2 83 E2 29
CBC-MAC : 6A DB 15 B6 71 81 B2 E2
Début CTR : 01 00 F8 B6 78 09 4E 3B 3C 96 96 76 6C FA 00 01
CTR[0001] : BD CE 95 5C CF D3 81 0A 91 EA 77 A6 A4 5B C0 4C
CTR[0002] : 43 2E F2 32 AE 36 D8 92 22 BF 63 37 E6 B2 6C E8
CTR[MAC] : 1C F7 19 C1 35 7F CC DE
Longueur totale de paquet = 40. [résultat authentifié et chiffré]
77 B6 0F 01 1C 03 E1 52 58 99 BC AE 55 45 FF 1A
08 5E E2 EF BF 52 B2 E0 4B EE 1E 23 36 C7 3E 3F
76 2C 0C 77 44 FE 7E 3C

========== Vecteur paquet n° 18 ==========
Clé AES = D7 82 8D 13 B2 B0 BD C3 25 A7 62 36 DF 93 CC 6B
Nom occasionnel = 00 D5 60 91 2D 3F 70 3C 96 96 76 6C FA

Longueur totale de paquet = 33. [Entré avec 12 octets d'en-tête de texte en clair]
CD 90 44 D2 B7 1F DB 81 20 EA 60 C0 64 35 AC BA
FB 11 A8 2E 2F 07 1D 7C A4 A5 EB D9 3A 80 3B A8
7F
IV CBC entrant : 59 00 D5 60 91 2D 3F 70 3C 96 96 76 6C FA 00 15
IV CBC sortant :BA 37 74 54 D7 20 A4 59 25 97 F6 A3 D1 D6 BA 67
Après OUx : BA 3B B9 C4 93 F2 13 46 FE 16 D6 49 B1 16 BA 67 [en-tête]
Après AES : 81 6A 20 20 38 D0 A6 30 CB E0 B7 3C 39 BB CE 05
Après OUx : E5 5F 8C 9A C3 C1 0E 1E E4 E7 AA 40 9D 1E 25 DC [message]
Après AES : 6D 5C 15 FD 85 2D 5C 3C E3 03 3D 85 DA 57 BD AC
Après OUx : 57 DC 2E 55 FA 2D 5C 3C E3 03 3D 85 DA 57 BD AC [message]
Après AES : B0 4A 1C 23 BC 39 B6 51 76 FD 5B FF 9B C1 28 5E
CBC-MAC : B0 4A 1C 23 BC 39 B6 51
Début CTR : 01 00 D5 60 91 2D 3F 70 3C 96 96 76 6C FA 00 01
CTR[0001] : 64 A2 C5 56 50 CE E0 4C 7A 93 D8 EE F5 43 E8 8E
CTR[0002] : 18 E7 65 AC B7 B0 E9 AF 09 2B D0 20 6C A1 C8 3C
CTR[MAC] : F7 43 82 79 5C 49 F3 00
Longueur totale de paquet = 41. [résultat authentifié et chiffré]
CD 90 44 D2 B7 1F DB 81 20 EA 60 C0 00 97 69 EC
AB DF 48 62 55 94 C5 92 51 E6 03 57 22 67 5E 04
C8 47 09 9E 5A E0 70 45 51

========== Vecteur paquet n° 19 ==========
Clé AES = D7 82 8D 13 B2 B0 BD C3 25 A7 62 36 DF 93 CC 6B
Nom occasionnel = 00 42 FF F8 F1 95 1C 3C 96 96 76 6C FA
Longueur totale de paquet = 31. [Entré avec 8 octets d'en-tête de texte en clair]
D8 5B C7 E6 9F 94 4F B8 8A 19 B9 50 BC F7 1A 01
8E 5E 67 01 C9 17 87 65 98 09 D6 7D BE DD 18
IV CBC entrant : 61 00 42 FF F8 F1 95 1C 3C 96 96 76 6C FA 00 17
IV CBC sortant :44 F7 CC 9C 2B DD 2F 45 F6 38 25 6B 73 6E 1D 7A
Après OUx : 44 FF 14 C7 EC 3B B0 D1 B9 80 25 6B 73 6E 1D 7A [en-tête]
Après AES : 57 C3 73 F8 00 AA 5F CC 7B CF 1D 1B DD BB 4C 52
Après OUx : DD DA CA A8 BC 5D 45 CD F5 91 7A 1A 14 AC CB 37 [message]
Après AES : 42 4E 93 72 72 C8 79 B6 11 C7 A5 9F 47 8D 9F D8
Après OUx : DA 47 45 0F CC 15 61 B6 11 C7 A5 9F 47 8D 9F D8 [message]
Après AES : 9A CB 03 F8 B9 DB C8 D2 D2 D7 A4 B4 95 25 08 67
CBC-MAC : 9A CB 03 F8 B9 DB C8 D2 D2 D7
Début CTR : 01 00 42 FF F8 F1 95 1C 3C 96 96 76 6C FA 00 01
CTR[0001] : 36 38 34 FA 28 83 3D B7 55 66 0D 98 65 0D 68 46
CTR[0002] : 35 E9 63 54 87 16 72 56 3F 0C 08 AF 78 44 31 A9
CTR[MAC] : F9 B7 FA 46 7B 9B 40 45 14 6D
Longueur totale de paquet = 41. [résultat authentifié et chiffré]
D8 5B C7 E6 9F 94 4F B8 BC 21 8D AA 94 74 27 B6
DB 38 6A 99 AC 1A EF 23 AD E0 B5 29 39 CB 6A 63
7C F9 BE C2 40 88 97 C6 BA

========== Vecteur paquet n° 20 ==========
Clé AES = D7 82 8D 13 B2 B0 BD C3 25 A7 62 36 DF 93 CC 6B
Nom occasionnel = 00 92 0F 40 E5 6C DC 3C 96 96 76 6C FA
Longueur totale de paquet = 32. [Entré avec 8 octets d'en-tête de texte en clair]
74 A0 EB C9 06 9F 5B 37 17 61 43 3C 37 C5 A3 5F
C1 F3 9F 40 63 02 EB 90 7C 61 63 BE 38 C9 84 37
IV CBC entrant : 61 00 92 0F 40 E5 6C DC 3C 96 96 76 6C FA 00 18
IV CBC sortant :60 CB 21 CE 40 06 50 AE 2A D2 BE 52 9F 5F 0F C2
Après OUx : 60 C3 55 6E AB CF 56 31 71 E5 BE 52 9F 5F 0F C2 [en-tête]
Après AES : 03 20 64 14 35 32 5D 95 C8 A2 50 40 93 28 DA 9B
Après OUx : 14 41 27 28 02 F7 FE CA 09 51 CF 00 F0 2A 31 0B [message]
Après AES : B9 E8 87 95 ED F7 F0 08 15 15 F0 14 E2 FE 0E 48
Après OUx : C5 89 E4 2B D5 3E 74 3F 15 15 F0 14 E2 FE 0E 48 [message]
Après AES : 8F AD 0C 23 E9 63 7E 87 FA 21 45 51 1B 47 DE F1
CBC-MAC : 8F AD 0C 23 E9 63 7E 87 FA 21
Début CTR : 01 00 92 0F 40 E5 6C DC 3C 96 96 76 6C FA 00 01

CTR[0001] : 4F 71 A5 C1 12 42 E3 7D 29 F0 FE E4 1B E1 02 5F
CTR[0002] : 34 2B D3 F1 7C B7 7B C1 79 0B 05 05 61 59 27 2C
CTR[MAC] : 7F 09 7B EF C6 AA C1 D3 73 65
Longueur totale de paquet = 42. [résultat authentifié et chiffré]
74 A0 EB C9 06 9F 5B 37 58 10 E6 FD 25 87 40 22
E8 03 61 A4 78 E3 E9 CF 48 4A B0 4F 44 7E FF F6
F0 A4 77 CC 2F C9 BF 54 89 44

========== Vecteur paquet n° 21 ==========
Clé AES = D7 82 8D 13 B2 B0 BD C3 25 A7 62 36 DF 93 CC 6B
Nom occasionnel = 00 27 CA 0C 71 20 BC 3C 96 96 76 6C FA
Longueur totale de paquet = 33. [Entré avec 8 octets d'en-tête de texte en clair]
44 A3 AA 3A AE 64 75 CA A4 34 A8 E5 85 00 C6 E4
15 30 53 88 62 D6 86 EA 9E 81 30 1B 5A E4 22 6B
FA
IV CBC entrant : 61 00 27 CA 0C 71 20 BC 3C 96 96 76 6C FA 00 19
IV CBC sortant :43 07 C0 73 A8 9E E1 D5 05 27 B2 9A 62 48 D6 D2
Après OUx : 43 0F 84 D0 02 A4 4F B1 70 ED B2 9A 62 48 D6 D2 [en-tête]
Après AES : B6 0B C6 F5 84 01 75 BC 01 27 70 F1 11 8D 75 10
Après OUx : 12 3F 6E 10 01 01 B3 58 14 17 23 79 73 5B F3 FA [message]
Après AES : 7D 5E 64 92 CE 2C B9 EA 7E 4C 4A 09 09 89 C8 FB
Après OUx : E3 DF 54 89 94 C8 9B 81 84 4C 4A 09 09 89 C8 FB [message]
Après AES : 68 5F 8D 79 D2 2B 9B 74 21 DF 4C 3E 87 BA 0A AF
CBC-MAC : 68 5F 8D 79 D2 2B 9B 74 21 DF
Début CTR : 01 00 27 CA 0C 71 20 BC 3C 96 96 76 6C FA 00 01
CTR[0001] : 56 8A 45 9E 40 09 48 67 EB 85 E0 9E 6A 2E 64 76
CTR[0002] : A6 00 AA 92 92 03 54 9A AE EF 2C CC 59 13 7A 57
CTR[MAC] : 25 1E DC DD 3F 11 10 F3 98 11
Longueur totale de paquet = 43. [résultat authentifié et chiffré]
44 A3 AA 3A AE 64 75 CA F2 BE ED 7B C5 09 8E 83
FE B5 B3 16 08 F8 E2 9C 38 81 9A 89 C8 E7 76 F1
54 4D 41 51 A4 ED 3A 8B 87 B9 CE

========== Vecteur paquet n° 22 ==========
Clé AES = D7 82 8D 13 B2 B0 BD C3 25 A7 62 36 DF 93 CC 6B
Nom occasionnel = 00 5B 8C CB CD 9A F8 3C 96 96 76 6C FA
Longueur totale de paquet = 31. [Entré avec 12 octets d'en-tête de texte en clair]
EC 46 BB 63 B0 25 20 C3 3C 49 FD 70 B9 6B 49 E2
1D 62 17 41 63 28 75 DB 7F 6C 92 43 D2 D7 C2
IV CBC entrant : 61 00 5B 8C CB CD 9A F8 3C 96 96 76 6C FA 00 13
IV CBC sortant :91 14 AD 06 B6 CC 02 35 76 9A B6 14 C4 82 95 03
Après OUx : 91 18 41 40 0D AF B2 10 56 59 8A 5D 39 F2 95 03 [en-tête]
Après AES : 29 BD 7C 27 83 E3 E8 D3 C3 5C 01 F4 4C EC BB FA
Après OUx : 90 D6 35 C5 9E 81 FF 92 A0 74 74 2F 33 80 29 B9 [message]
Après AES : 4E DA F4 0D 21 0B D4 5F FE 97 90 B9 AA EC 34 4C
Après OUx : 9C 0D 36 0D 21 0B D4 5F FE 97 90 B9 AA EC 34 4C [message]
Après AES : 21 9E F8 90 EA 64 C2 11 A5 37 88 83 E1 BA 22 0D
CBC-MAC : 21 9E F8 90 EA 64 C2 11 A5 37
Début CTR : 01 00 5B 8C CB CD 9A F8 3C 96 96 76 6C FA 00 01
CTR[0001] : 88 BC 19 42 80 C1 FA 3E BE FC EF FB 4D C6 2D 54
CTR[0002] : 3E 59 7D A5 AE 21 CC A4 00 9E 4C 0C 91 F6 22 49
CTR[MAC] : 5C BC 30 98 66 02 A9 F4 64 A0
Longueur totale de paquet = 41. [résultat authentifié et chiffré]
EC 46 BB 63 B0 25 20 C3 3C 49 FD 70 31 D7 50 A0
9D A3 ED 7F DD D4 9A 20 32 AA BF 17 EC 8E BF 7D
22 C8 08 8C 66 6B E5 C1 97

========== Vecteur paquet n° 23 ==========
Clé AES = D7 82 8D 13 B2 B0 BD C3 25 A7 62 36 DF 93 CC 6B
Nom occasionnel = 00 3E BE 94 04 4B 9A 3C 96 96 76 6C FA
Longueur totale de paquet = 32. [Entré avec 12 octets d'en-tête de texte en clair]
47 A6 5A C7 8B 3D 59 42 27 E8 5E 71 E2 FC FB B8

```
        80 44 2C 73  1B F9 51 67  C8 FF D7 89  5E 33 70 76
IV CBC entrant : 61 00 3E BE  94 04 4B 9A  3C 96 96 76  6C FA 00 14
IV CBC sortant :0F 70 3F 5A  54 2C 44 6E  8B 74 A3 73  9B 48 B9 61
Après OUx : 0F 7C 78 FC  0E EB CF 53  D2 36 84 9B  C5 39 B9 61   [en-tête]
Après AES : 40 5B ED 29  D0 98 AE 91  DB 68 78 F3  68 B8 73 85
Après OUx : A2 A7 16 91  50 DC 82 E2  C0 91 29 94  A0 47 A4 0C   [message]
Après AES : 3D 03 29 3C  FD 81 1B 37  01 51 FB C7  85 6B 7A 74
Après OUx : 63 30 59 4A  FD 81 1B 37  01 51 FB C7  85 6B 7A 74   [message]
Après AES : 66 4F 27 16  3E 36 0F 72  62 0D 4E 67  7C E0 61 DE
CBC-MAC  : 66 4F 27 16  3E 36 0F 72  62 0D
Début CTR : 01 00 3E BE  94 04 4B 9A  3C 96 96 76  6C FA 00 01
CTR[0001]: 0A 7E 0A 63  53 C8 CF 9E  BC 3B 6E 63  15 9A D0 97
CTR[0002]: EA 20 32 DA  27 82 6E 13  9E 1E 72 5C  5B 0D 3E BF
CTR[MAC ]: B9 31 27 CA  F0 F1 A1 20  FA 70
Longueur totale de paquet = 42. [résultat authentifié et chiffré]
        47 A6 5A C7  8B 3D 59 42  27 E8 5E 71  E8 82 F1 DB
        D3 8C E3 ED  A7 C2 3F 04  DD 65 07 1E  B4 13 42 AC
        DF 7E 00 DC  CE C7 AE 52  98 7D

  ================ Vecteur paquet n° 24 ==================
Clé AES =  D7 82 8D 13  B2 B0 BD C3  25 A7 62 36  DF 93 CC 6B
Nom occasionnel = 00 8D 49 3B  30 AE 8B 3C  96 96 76 6C  FA
Longueur totale de paquet = 33. [Entré avec 12 octets d'en-tête de texte en clair]
        6E 37 A6 EF  54 6D 95 5D  34 AB 60 59  AB F2 1C 0B
        02 FE B8 8F  85 6D F4 A3  73 81 BC E3  CC 12 85 17
        D4
IV CBC entrant : 61 00 8D 49  3B 30 AE 8B  3C 96 96 76  6C FA 00 15
IV CBC sortant :67 AC E4 E8  06 77 7A D3  27 1D 0B 93  4C 67 98 15
Après OUx : 67 A0 8A DF  A0 98 2E BE  B2 40 3F 38  2C 3E 98 15   [en-tête]
Après AES : 35 58 F8 7E  CA C2 B4 39  B6 7E 75 BB  F1 5E 69 08
Après OUx : 9E AA E4 75  C8 3C 0C B6  33 13 81 18  82 DF D5 EB   [message]
Après AES : 54 E4 7B 62  22 F0 BB 87  17 D0 71 6A  EB AF 19 9E
Après OUx : 98 F6 FE 75  F6 F0 BB 87  17 D0 71 6A  EB AF 19 9E   [message]
Après AES : 23 E3 30 50  BC 57 DC 2C  3D 3E 7C 94  77 D1 49 71
CBC-MAC : 23 E3 30 50  BC 57 DC 2C  3D 3E
Début CTR : 01 00 8D 49  3B 30 AE 8B  3C 96 96 76  6C FA 00 01
CTR[0001] : 58 DB 19 B3  88 9A A3 8B  3C A4 0B 16  FF 42 2C 73
CTR[0002] : C3 2F 24 3D  65 DC 7E 9F  4B 02 16 AB  7F B9 6B 4D
CTR[MAC] : 4E 2D AE D2  53 F6 B1 8A  1D 67
Longueur totale de paquet = 43. [résultat authentifié et chiffré]
        6E 37 A6 EF  54 6D 95 5D  34 AB 60 59  F3 29 05 B8
        8A 64 1B 04  B9 C9 FF B5  8C C3 90 90  0F 3D A1 2A
        B1 6D CE 9E  82 EF A1 6D  A6 20 59
```

9. Déclarations de propriété intellectuelle

Les auteurs renoncent explicitement à tous droits de propriété intellectuelle sur CCM au profit du domaine public. De plus, les auteurs ne sont pas informés de brevets ou d'application de brevet qui couvrent le mode CCM quelque part dans le monde. Il est estimé que CCM est une simple combinaison de techniques bien établies, et on estime que CCM est évident pour toute personne d'une capacité ordinaire.

10. Considérations sur la sécurité

On prétend que ce mode de chiffrement de bloc est sûr contre les attaquants limités à 2^128 étapes d'opération si la clé K fait 256 bits ou plus. Il y a des attaques très générique de calcul préalable contre tous les modes de chiffrement de bloc qui permettent une attaque de moyen terme sur la clé K. Si ces attaques peuvent être faites, leur force théorique, et toute autre, le mode de chiffrement est limité à 2^(n/2) où n est le nombre de bits de la clé. La force de l'authentification est bien sûr limitée par M.

Les utilisateurs de plus petites tailles de clés (comme des clés de 128 bits) devraient prendre des précautions pour rendre les attaques de calcul préalable plus difficiles. L'utilisation répétée de la même valeur de nom occasionnel (avec bien sûr des clés différentes) devrait être évitée. Une solution est d'inclure une valeur aléatoire dans le nom occasionnel. Bien sûr, un compteur de paquets est aussi nécessaire au sein du nom occasionnel. Comme le nom occasionnel est de taille limitée, une valeur aléatoire dans le nom occasionnel fournit une quantité limitée de sécurité supplémentaire.

11. Références

Cette section donne les références normatives et les références pour information.

11.1 Références normatives

[RFC2119] S. Bradner, "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997. *(MàJ par* RFC8174)

11.2 Références pour information

[AES] NIST, FIPS PUB 197, "Advanced Encryption Standard (AES)," November 2001.

[CCM] Whiting, D., Housley, R. and N. Ferguson, "AES Encryption & Authentication Using CTR Mode & CBC-MAC," IEEE P802.11 doc 02/001r2, mai 2002.

[ESP] Kent, S. and R. Atkinson, "IP Encapsulating Security Payload (ESP)", RFC 2406, novembre 1998.

[MAC] NIST, FIPS PUB 113, "Computer Data Authentication," mai 1985.

[MODES] Dworkin, M., "Recommendation for Block Cipher Modes of Operation: Methods and Techniques," NIST Special Publication 800-38A, décembre 2001.

[OCB] Rogaway, P., Bellare, M., Black, J. and T, Krovetz, "OCB: A block-Cipher Mod of Operation for Efficient Authenticated Encryption," 8th ACM Conference on Computer and Communications Security, pp 196-295, ACM Press, 2001.

[PROOF] Jonsson, J., "On the Security of CTR + CBC-MAC," SAC 2002 -- Ninth Annual Workshop on Selected Areas of Cryptography, Workshop Record version, 2002. Version finale à paraître dans les LNCS Proceedings.

12. Remerciements

Russ Housley remercie la direction de RSA Laboratories, et en particulier Burt Kaliski, qui a soutenu le développement de ce mode cryptographique et la présente spécification. La grande majorité de ce travail a été faite alors que Russ était employé de RSA Laboratories.

13. Adresse des auteurs

Doug Whiting
Hifn
5973 Avenida Encinas, #110
Carlsbad, CA 92009
USA
mél : dwhiting@hifn.com

Russell Housley
Vigil Security, LLC
918 Spring Knoll Drive
Herndon, VA 20170
USA
mél: housley@vigilsec.com

Niels Ferguson
MacFergus BV
Bart de Ligtstraat 64
1097 JE Amsterdam
Netherlands
mél : niels@macfergus.com

14. Déclaration complète de droits de reproduction

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