Groupe de travail Réseau
Request for Comments : 2451
Catégorie : En cours de normalisation
Traduction Claude Brière de L'Isle

R. Pereira, TimeStep Corporation
R. Adams, Cisco Systems Inc.
November 1998

Algorithmes de chiffrement ESP en mode CBC

Résumé
Le présent document décrit comment utiliser les algorithmes de chiffrement en mode CBC avec le protocole d'encapsulation de charge utile de sécurité (ESP, *Encapsulating Security Payload*) IPSec. Il explique clairement non seulement comment utiliser certains algorithmes de chiffrement, mais aussi comment utiliser tous les algorithmes de chiffrement en mode CBC.

Table des matières

1. Introduction

L'encapsulation de charge utile de sécurité (ESP, *Encapsulating Security Payload*) [RFC2406] assure la confidentialité aux datagrammes IP en chiffrant les données de la charge utile à protéger. La présente spécification décrit l'utilisation par ESP des algorithmes de chiffrement en mode CBC.

Comme ce document ne décrit pas l'utilisation de l'algorithme de chiffrement DES par défaut, le lecteur devrait être familiarisé avec la [RFC2405].

On suppose que les termes et concepts décrits dans les documents "Architecture de sécurité pour le protocole Internet" [RFC2401], "Feuille de route pour la sécurité sur IP" [RFC2411], et "Encapsulation de charge utile de sécurité (ESP) IP" [RFC2406] sont familiers au lecteur.

De plus, le présent document accompagne la [RFC2406] et DOIT être lu dans ce contexte.

1.1 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].

1.2 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 pourraient 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.

2. Algorithmes de chiffrement

Tous les algorithmes de chiffrement de bloc symétrique partagent des caractéristiques et variables communes. Cela inclut le mode, la taille de clé, les clés faibles, la taille de bloc, et les tours. Tous sont expliqués ci-dessous.

Bien que le présent document illustre certains algorithmes de chiffrement tels que Blowfish [Schneier93], CAST-128 [RFC2144], 3DES, IDEA [Lai] [MOV], et RC5 [RFC2040], tout autre algorithme de chiffrement de bloc peut être utilisé avec ESP si toutes les variables décrites dans le présent document sont clairement définies.

2.1 Mode

Tous les algorithmes de chiffrement de bloc symétrique décrits ou évoqués au sein de ce document utilisent le mode de chaînage de bloc de chiffrement (CBC, *Cipher Block Chaining*). Ce mode requiert un vecteur d'initialisation (IV, *Initialization Vector*) qui est de la même taille que celle du bloc. L'utilisation d'un IV généré de façon aléatoire empêche la génération de texte chiffré identique à partir de paquets qui ont des données identiques dépassant le premier bloc de la taille de bloc de l'algorithme de chiffrement.

Le IV est soumis à l'opération OUX avec le premier bloc de texte en clair, avant d'être chiffré. Puis pour les blocs suivants, le bloc de texte chiffré précédent est passé à l'opération OUX avec le bloc de texte en clair actuel, avant d'être chiffré.

Plus d'informations sur le mode CBC seront obtenues dans [Schneier95].

2.2 Taille de clé

Certains algorithmes de chiffrement permettent des clés de taille variable, tandis que d'autres ne permettent qu'une taille de clé spécifique. La longueur de la clé est corrélée avec la force de l'algorithme, et donc de plus longues clés sont toujours plus difficiles à casser que des plus courtes.

Le présent document stipule que toutes les tailles de clé DOIVENT être un multiple de 8 bits.

Le présent document spécifie la taille de clé par défaut pour chaque algorithme de chiffrement. Cette taille a été choisie après consultation des experts de ces algorithmes et en mettant en balance la force de l'algorithme avec ses performances.

Algorithme	Tailles de clé (bits)	Tailles courantes	Par défaut
CAST-128 [1]	40 à 128	40, 64, 80, 128	128
RC5	40 à 2040	40, 128, 160	128
IDEA	128	128	128
Blowfish	40 à 448	128	128
3DES [2]	192	192	192

Notes :

[1] Avec CAST-128, les clés de moins de 128 bits DOIVENT être bourrées avec des zéros dans les positions les plus à droite, ou de moindre poids, des 128 bits car le programme de clés CAST-128 suppose une entrée de clé de 128 bits. Donc, si on a une clé d'une taille de 80 bits '3B5D831CFE', elle sera bourrée pour produire une clé avec une taille de 128 bits '3B5D831CFE000000'.

[2] La première clé 3DES est prise des 64 premiers bits, la seconde des 64 bits suivants, et la troisième des 64 derniers bits. Les mises en œuvre DOIVENT prendre en considération les bits de parité lorsque elles acceptent initialement un nouvel ensemble de clés. Chacune des trois clés est en réalité longue de 56 bits avec les 8 bits supplémentaires utilisés pour la parité.

Le lecteur devrait noter que la taille de clé minimum pour tous les algorithmes de chiffrement ci-dessus est de 40 bits, et que les auteurs recommandent fortement que les mises en œuvre N'UTILISENT PAS de taille de clé inférieure à 40 bits.

2.3 Clés faibles

Des vérifications de la force des clés DEVRAIENT être effectuées. Si une clé faible est trouvée, elle DEVRAIT être rejetée et une nouvelle SA demandée. Certains algorithmes de chiffrement ont une ou des clés faibles qui NE DOIVENT PAS être utilisées à cause de leur nature faible.

De nouvelles clés faibles peuvent être découvertes, de sorte que le présent document ne contient en aucun cas toutes les clés faibles possibles pour ces chiffres. Prière de vérifier avec d'autres sources de cryptographie telles que [MOV] et [Schneier] d'autres clés faibles.

CAST-128 :
Pas de clé faible connue.

RC5 :
Pas de clé faible connue quand il est utilisé avec 16 tours.

IDEA :
Il a été trouvé que IDEA a des clés faibles. Prière de regarder dans [MOV] et [Schneier] pour plus d'informations.

Blowfish :
Des clés faibles ont été découvertes pour Blowfish. Les clés faibles sont des clés qui produisent des entrées identiques dans une certaine S-box. Malheureusement, il n'y a pas de moyen d'essayer la force des clés avant que les valeurs de la S-box ne soient générées. Cependant, les chances de générer de façon aléatoire une telle clé sont faibles.

3DES :
DES a 64 clés faible connues, y compris les soi-disant clés semi faibles et les clés éventuellement faibles [Schneier95, pp 280-282]. La probabilité de tomber dessus au hasard est négligeable.

Pour DES-EDE3, il n'y a pas de besoin connu de rejeter des clés faibles ou de complément. Toute faiblesse est évitée par l'utilisation de plusieurs clés.

Cependant, si les deux premières clés ou les deux dernières clés indépendantes de 64 bits sont égales (k1 == k2 ou k2 == k3) alors le fonctionnement de 3DES est simplement le même que celui de DES. Les mises en œuvre DOIVENT rejeter les clés qui possèdent cette propriété.

2.4 Taille de bloc et bourrage

Tous les algorithmes décrits dans le présent document utilisent une taille de bloc de huit octets (64 bits).

Le bourrage est utilisé pour aligner les octets de type de charge utile et de longueur de bourrage comme spécifié dans la [RFC2406]. Le bourrage doit être suffisant pour aligner les données à chiffrer sur une limite de huit octets (64 bits).

2.5 Tours

Cette variable détermine combien de fois un bloc est chiffré. Bien que cette variable PUISSE être négociée, une valeur par défaut DOIT toujours exister lorsque elle n'est pas négociée.

Algorithme	**Négociable**	**Tours par défaut**
CAST-128	Non	clé ≤ 80 bits, 12
RC5	Non	16
IDEA	Non	8
Blowfish	Non	16
3DES	Non	48 (16 x 3)

2.6 Cadre

CAST-128 :
La procédure de conception de CAST a été développée à l'origine par Carlisle Adams et Stafford Tavares à l'Université de la Reine à Kingston, Ontario, Canada. Les améliorations ultérieures ont été apportées au fil des ans par Carlisle Adams et Michael Wiener de Entrust Technologies. CAST-128 est le résultat de l'application de la procédure de conception de CAST comme expliqué dans la [RFC2144].

RC5 :
L'algorithme de chiffrement RC5 a été développé par Ron Rivest pour RSA Data Security Inc. afin de satisfaire les besoins d'une solution de remplacement de logiciel et matériel de chiffrement à hautes performances à DES. Il est breveté (brevet n° 5 724 428). Une description de RC5 se trouve dans [MOV] et dans [Schneier].

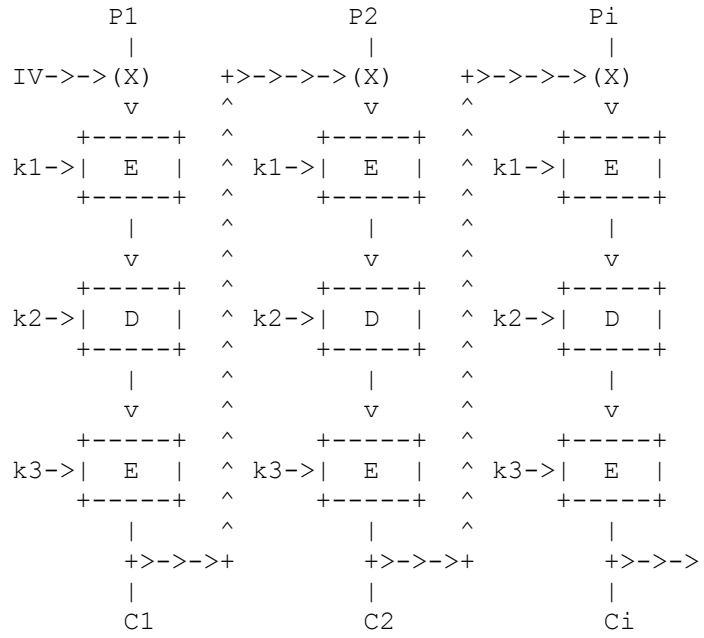
IDEA :
Xuejia Lai et James Massey ont développé l'algorithme IDEA (International Data Encryption Algorithm, *algorithme international de chiffrement de données*). L'algorithme est décrit en détail dans [Lai], [Schneier] et [MOV].

L'algorithme IDEA est breveté en Europe et aux États Unis et une demande de brevet est en cours au Japon. Des licences sont exigées pour les utilisations commerciales de IDEA.

Pour les informations sur le brevet et les licences, contacter :
Ascom Systec AG, Dept. CMVV
Gewerbepark, CH-5506
Magenwil, Confédération Helvétique
téléphone : +41 64 56 59 83
Fax : +41 64 56 59 90
idea@ascom.ch
http://www.ascom.ch/Web/systec/policy/normal/exhibit1.html

Blowfish :
Bruce Schneier de Counterpane Systems a développé l'algorithme de chiffrement de bloc Blowfish. L'algorithme est décrit en détails dans [Schneier93], [Schneier95] et [Schneier].

3DES :
Cette variante de DES, appelée familièrement "Triple DES" ou DES-EDE3, traite chaque bloc trois fois, chaque fois avec une clé différente. Cette technique d'utiliser plus d'une opération DES a été proposée dans [Tuchman79].



L'algorithme DES-EDE3-CBC est une simple variante de l'algorithme DES-CBC [FIPS-46]. La technique de chaînage "externe" est utilisée.

Dans DES-EDE3-CBC, un vecteur d'initialisation (IV) est passé à l'opération OUX avec le premier bloc de 64 bits (8 octets) de texte en clair (P1). La fonction DES à clé est itérée trois fois : un chiffrement (Ek1) suivi par un déchiffrement (Dk2) suivi par un chiffrement (Ek3) et génère le texte chiffré (C1) pour le bloc. Chaque itération utilise une clé indépendante : k1, k2 et k3.

Pour les blocs successifs, le bloc précédent de texte chiffré est passé à l'opération OUX avec le texte en clair actuel (Pi). La fonction de chiffrement DES-EDE3 à clés génère le texte chiffré (Ci) pour ce bloc.

Pour déchiffrer, l'ordre des fonctions est inversé : déchiffrement avec k3, chiffrement avec k2, déchiffrement avec k1, et OUX avec le précédent bloc de texte chiffré.

Noter que lorsque les trois clés (k1, k2 et k3) sont la même, DES-EDE3-CBC est équivalent à DES-CBC. Cette propriété permet aux mises en œuvre de matériels DES-EDE3 de fonctionner en mode DES sans modification.

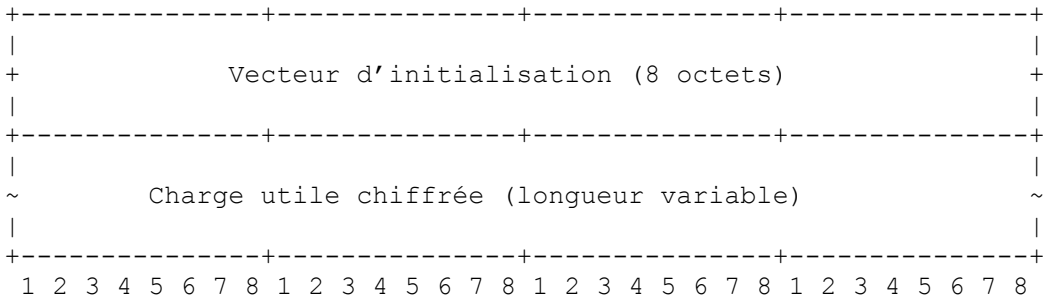
Pour plus d'explications et des informations sur la mise en œuvre de Triple DES, voir [Schneier95].

2.7 Performances

Pour un tableau de comparaison des vitesses estimées de ces algorithmes de chiffrement et des autres, prière de voir [Schneier97] ou pour une mise à jour de la comparaison des performances, voir [Bosseleaers].

3. Charge utile ESP

La charge utile ESP est constituée de l'IV suivi par le texte chiffré brut. Donc, le champ charge utile, comme défini dans la [RFC2406], est divisé selon le diagramme suivant



Le champ IV DOIT être de la même taille que le bloc de l'algorithme chiffrement utilisé. L'IV DOIT être choisi au hasard. La pratique courante est d'utiliser des données aléatoires pour le premier IV et le dernier bloc de données chiffrées provenant d'un processus de chiffrement comme IV pour le prochain processus de chiffrement.

Inclure le vecteur d'initialisation dans chaque datagramme assure que le déchiffrement de chaque datagramme reçu peut être effectué, même lorsque certains datagrammes sont abandonnés, ou que des datagrammes sont déclassés dans le transit.

Pour éviter le chiffrement ECB de blocs de texte en clair très similaires dans des paquets différents, les mises en œuvre NE DOIVENT PAS utiliser un compteur ou autre source de distance de Hamming faible pour les IV.

3.1 Considérations sur l'environnement de ESP

Il n'y a actuellement, pas de problème connu concernant les interactions entre ces algorithmes et d'autres aspects de ESP, tels que l'utilisation de certains schémas d'authentification.

3.2 Matériel de clés

Le nombre minimum de bits envoyés à partir du protocole d'échange de clés pour cet algorithme ESP doit être supérieur ou égal à la taille de clé.

La clé de chiffrement et de déchiffrement du chiffre est tirée des <x> premiers bits du matériel de clé, où <x> représente la taille de clé requise.

4. Considérations pour la sécurité

Les mises en œuvre sont invitées à utiliser les plus grandes tailles de clé qu'elles peuvent lorsque elles prennent en compte les considérations de performances pour leur configuration particulière de matériel et logiciel. Noter que le chiffrement impacte nécessairement les deux côtés d'un canal sûr, de sorte qu'une certaine considération doit être apportée non seulement au côté client, mais aussi au côté serveur.

Pour des informations sur les cas où utiliser des valeurs aléatoires, prière de voir [Bell97].

Pour plus de considérations sur la sécurité, le lecteur est invité à lire les documents qui décrivent les algorithmes de chiffrement eux-mêmes.

5. Références

[Bell97] S. Bellovin, "Probable Plaintext Cryptanalysis of the IP Security Protocols", Compte-rendu du Symposium on Network and Distributed System Security, San Diego, CA, pp. 155-160, février 1997 (aussi à http://www.research.att.com/~smb/probtxt .{ps, pdf}).

[Bosselaers] A. Bosselaers, "Performance of Pentium implementations", http://www.esat.kuleuven.ac.be/~bosselae/

[Crypto93] J. Daemen, R. Govaerts, J. Vandewalle, "Weak Keys for IDEA", Advances in Cryptology, CRYPTO 93 Proceedings, Springer-Verlag, pp. 224-230.

[FIPS-46] US National Bureau of Standards, "Data Encryption Standard", Federal Information Processing Standard (FIPS) Publication 46, janvier 1977.

[Lai] X. Lai, "On the Design and Security of Block Ciphers", ETH Series in Information Processing, v. 1, Konstanz: Hartung-Gorre Verlag, 1992.

[RFC2040] R. Baldwin et R. Rivest, "Algorithmes RC5, RC5-CBC, RC5-CBC-Pad, et RC5-CTS", octobre 1996. *(Information)*

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

[RFC2144] C. Adams, "L'algorithme de chiffrement CAST-128", mai 1997. *(Information)*

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

[RFC2405] C. Madson et N. Doraswamy, "Algorithme de chiffrement ESP DES-CBC avec IV explicite", novembre 1998.

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

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

[MOV] A. Menezes, P. Van Oorschot, S. Vanstone, "Handbook of Applied Cryptography", CRC Press, 1997. ISBN 0-8493- 8523-7

[Schneier] B. Schneier, "Applied Cryptography Second Edition", John Wiley & Sons, New York, NY, 1995. ISBN 0-471-12845-7

[Schneier93] B. Schneier, "Description of a New Variable-Length Key, 64-Bit Block Cipher", d'après "Fast Software Encryption, Cambridge Security Workshop Proceedings", Springer-Verlag, 1994, pp. 191-204. http://www.counterpane.com/bfsverlag.html

[Schneier95] B. Schneier, "The Blowfish Encryption Algorithm - One Year Later", Dr. Dobb's Journal, septembre 1995, http://www.counterpane.com/bfdobsoyl.html

[Schneier97] B. Scheier, "Speed Comparisons of Block Ciphers on a Pentium." f'vrier 1997, http://www.counterpane.com/speed.html

[Tuchman] Tuchman, W, "Hellman Presents No Shortcut Solutions to DES", IEEE Spectrum, v. 16 n. 7, juillet 1979, pp. 40-41.

6. Remerciements

Le présent document est une fusion de la plupart des documents des algorithmes de chiffrement d'ESP. Cette fusion a été faite pour faciliter une meilleure compréhension de ce qu'ont en commu tous les algorithmes ESP et pour favoriser le développement de ces algorithmes au sein d'ESP.

Le contenu de ce document se fonde sur des suggestions dont Stephen Kent est à l'origine et des discussions ultérieures sur la liste de diffusion IPSec ainsi que sur d'autres documents IPsec.

Des remerciements tout particuliers à Carlisle Adams et Paul Van Oorschot tous deux de Entrust Technologies qui ont fourni les éléments de CAST et les ont révisé.

Merci à tous les éditeurs des documents 3DES ESPprécédents ; W. Simpson, N. Doraswamy, P. Metzger, et P. Karn.

Merci à Brett Howard de TimeStep pour son travail original sur ESP-RC5.

Merci à Markku-Juhani Saarinen, Helger Lipmaa et Bart Preneel pour leur contribution sur IDEA et autres chiffres.

7. Adresses des éditeurs

Roy Pereira
TimeStep Corporation
téléphone : +1 (613) 599-3610 x 4808
mél : rpereira@timestep.com

Rob Adams
Cisco Systems Inc.
téléphone : +1 (408) 457-5397
mél : adams@cisco.com

Contributeurs :

Robert W. Baldwin
RSA Data Security, Inc.
télphone : +1 (415) 595-8782
mél : baldwin@rsa.com

Greg Carter
Entrust Technologies
téléphone : +1 (613) 763-1358

Rodney Thayer
Sable Technology Corporation
téléphone : +1 (617) 332-7292
mél : rodney@sabletech.com

Le groupe de travail IPSec peut être contacté la liste de diffusion du groupe de travail IPSec (ipsec@tis.com) ou par ses présidents :

Robert Moskowitz
International Computer Security Association
mél : rgm@icsa.net

Theodore Y. Ts'o
Massachusetts Institute of Technology
mél : tytso@MIT.EDU

8. Déclaration complète de droits de reproduction