Groupe de travail Réseau
Request for Comments : 3173
RFC rendue obsolète : 2393
Catégorie : En cours de normalisation
Traduction Claude Brière de L'Isle

A. Shacham, Juniper
B. Monsour, Consultant
R. Pereira, Cisco
M. Thomas, Consultant
septembre 2001

Protocole de compression de charge utile IP (IPComp)

Résumé
Le présent document décrit un protocole destiné à fournir une compression sans perte pour les datagrammes du protocole Internet dans un environnement Internet.

1. Introduction

La compression de charge utile IP est un protocole pour réduire la taille des datagrammes IP. Ce protocole va augmenter les performances globales de communication entre une paire d'hôtes/passerelles communicantes ("les nœuds") en compressant les datagrammes, pourvu que les nœuds aient une puissance de calcul suffisante, soit par la capacité du CPU, soit par un coprocesseur de compression, et que la communication soit sur des liaisons lentes ou encombrées.

La compression de charge utile IP est particulièrement utile lorsque le chiffrement est appliqué aux datagrammes IP. Chiffrer les datagrammes IP amène les données à être de nature aléatoire, rendant inefficace la compression à des couches de protocole inférieures (par exemple, le protocole PPP de contrôle de compression de la [RFC1962]). Si le chiffrement et la compression sont tous deux requis, la compression doit être appliquée avant le chiffrement.

Le présent document définit le protocole de compression de charge utile IP (IPComp), la structure de paquet IPComp, l'association IPComp (IPCA), et plusieurs méthodes de négociation de l'IPCA.

D'autres documents devront spécifier comment un algorithme spécifique de compression peut être utilisé avec le protocole de compression de charge utile IP. De tels algorithmes sont en dehors du domaine d'application du présent document.

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].

2. Processus de compression

Le traitement de compression des datagrammes IP a deux phases : la compression des datagrammes IP sortants ("compression") et la décompression des datagrammes entrants ("décompression"). Le traitement de compression DOIT être sans perte, assurant que le datagramme IP, après avoir été compressé et décompressé, est identique au datagramme IP d'origine.

Chaque datagramme IP est compressé et décompressé par lui-même sans aucune relation avec d'autres datagrammes ("compression sans état") car les datagrammes IP peuvent arriver décalés, ou ne pas arriver du tout. Chaque datagramme IP compressé encapsule une seule charge utile IP.

Le traitement des datagrammes IP entrants DOIT prendre en charge les datagrammes IP aussi bien compressés que non

compressés, afin de satisfaire aux exigences des politiques de non expansion, comme défini au paragraphe 2.2.

La compression des datagrammes IP sortants DOIT être faite avant tout traitement de sécurité IP, comme le chiffrement et l'authentification, et avant toute fragmentation du datagramme IP. De plus, dans IP version 6 [RFC2460], la compression des datagrammes IP sortants DOIT être faite avant l'ajout de l'en-tête Options bond par bond ou d'un en-tête d'acheminement, car tous deux portent des informations qui doivent être examinées et traitées par éventuellement tous les nœuds le long du chemin de livraison d'un paquet, et donc DOIVENT être envoyés sous la forme originale.

De même, la décompression des datagrammes IP entrants DOIT être faite après le réassemblage des datagrammes IP, et après l'achèvement de tous les traitements de sécurité IP, comme l'authentification et le déchiffrement.

2.1 Charge utile compressée

La compression est appliquée à un seul arrangement d'octets, qui sont contigus dans le datagramme IP. Cet arrangement d'octets se termine toujours au dernier octet de la charge utile du paquet IP. Noter qu'un arrangement contigu d'octets dans le datagramme IP peut n'être pas contigu dans la mémoire physique.

Dans IP version 4 [RFC0791], la compression est appliquée à la charge utile du datagramme IP, commençant au premier octet qui suit l'en-tête IP, et se poursuivant jusqu'an dernier octet du datagramme. Aucune portion de l'en-tête IP ou des options de l'en-tête IP n'est compressée. Noter que dans le cas d'un en-tête IP encapsulé (par exemple, encapsulation en mode tunnel dans IPsec) la charge utile du datagramme est définie comme commençant immédiatement après l'en-tête IP externe ; en conséquence, l'en-tête IP interne est considéré comme faisant partie de la charge utile et est compressé.

Dans le contexte IPv6, IPComp est vu comme une charge utile de bout en bout, et NE DOIT PAS s'appliquer aux en-têtes bond par bond, d'acheminement, et d'extension de fragmentation. La compression s'applique en commençant au premier champ Option d'en-tête IP qui ne porte pas d'informations qui doivent être examinées et traitées par les nœuds le long du chemin de livraison d'un paquet, si il existe un tel champ Option d'en-tête IP, et se continue avec la charge utile ULP du datagramme IP.

La taille d'une charge utile compressée, générée par l'algorithme de compression, DOIT être en unités d'octets complets.

Comme défini à la section 3, un en-tête IPComp est inséré immédiatement devant la charge utile compressée. L'en-tête IP d'origine est modifié pour indiquer l'usage du protocole IPComp et la réduction de taille du datagramme IP. Le contenu du champ Prochain en-tête (IPv6) ou protocole (IPv4) est mémorisé dans l'en-tête IPComp.

La décompression est appliquée à un seul arrangement contigu d'octets dans le datagramme IP. Le début de l'arrangement d'octets suit immédiatement l'en-tête IPComp et se termine au dernier octet de la charge utile IP. Si le processus de décompression s'achève avec succès, l'en-tête IP est modifié pour indiquer la taille du datagramme IP décompressé, et le prochain en-tête original tel que mémorisé dans l'en-tête IPComp. L'en-tête IPComp est retiré du datagramme IP et la charge utile décompressée suit immédiatement l'en-tête IP.

2.2 Politique de non expansion

Si la taille totale de la charge utile compressée et de l'en-tête IPComp, telle que définie à la section 3, n'est pas inférieure à la taille de la charge utile originale, le datagramme IP DOIT être envoyé sous la forme originale non compressée. Pour préciser : si un datagramme IP est envoyé non compressé, aucun en-tête IPComp n'est ajouté au datagramme. Cette politique assure l'économie des cycles de traitement de décompression et évite la fragmentation des datagrammes IP en cause lorsque le datagramme développé est supérieur à la MTU.

Les petits datagrammes IP vont vraisemblablement s'augmenter par suite de la compression. Donc, un seuil numérique devrait être appliqué avant la compression, afin que les datagrammes IP de taille inférieure au seuil soient envoyés sous la forme d'origine sans tenter la compression. Le seuil numérique est fonction de la mise en œuvre.

Un datagramme IP avec une charge utile qui a été précédemment compressée tend à ne plus se compresser. La charge utile précédemment compressée peut être le résultat d'un processus externe, tel qu'une compression appliquée par une couche supérieure dans la pile de communication, ou par un utilitaire de compression hors ligne. Un algorithme adaptatif devrait être mis en œuvre pour éviter une chute des performances. Par exemple, si la compression de i datagrammes IP consécutifs d'une IPCA échoue, les datagrammes IP suivants, disons k, sont envoyés sans tenter la compression. Si les j datagrammes suivants échouent aussi à la compression, un plus grand nombre de datagrammes, disons $k + n$, sera envoyé sans tenter la compression. Une fois qu'un datagramme a réussi à être compressé, le processus normal de IPComp recommence. Un tel

algorithme adaptatif, y compris tous les seuils qui s'y rapportent, dépend de la mise en œuvre.

Durant le traitement de la charge utile, l'algorithme de compression PEUT appliquer périodiquement un essai pour déterminer la compressibilité des données traitées, similaire à celui exigé par [V42BIS]. La nature de l'essai dépend de l'algorithme. Une fois que l'algorithme de compression a détecté que les données sont non compressibles, l'algorithme DEVRAIT arrêter le traitement des données, et la charge utile être envoyée sous la forme originale non compressée.

3. Structure d'en-tête de datagramme IP compressé

Un datagramme IP compressé est encapsulé en modifiant l'en-tête IP et en insérant un en-tête IPComp précédant immédiatement la charge utile compressée. La présente section définit les modifications de l'en-tête IP à la fois dans IPv4 et IPv6, et la structure de l'en-tête IPComp.

3.1 Modifications d'en-tête Ipv4

Les champs d'en-tête IPv4 suivants sont établis avant de transmettre le datagramme IP compressé :

Longueur totale
C'est la longueur du datagramme IP encapsulé entier, y compris l'en-tête IP, l'en-tête IPComp et la charge utile compressée.

Protocole
Le champ Protocole est réglé à 108, datagramme IPComp, [RFC1700].

Somme de contrôle d'en-tête
C'est la somme de contrôle d'en-tête Internet [RFC0791] de l'en-tête IP.

Tous les autres champs d'en-tête IPv4 restent inchangés, y compris toutes les options d'en-tête.

3.2 Modifications d'en-tête Ipv6

Les champs d'en-tête IPv6 suivants sont réglés avant de transmettre le datagramme IP compressé :

Longueur de charge utile
C'est la longueur de la charge utile IP compressée.

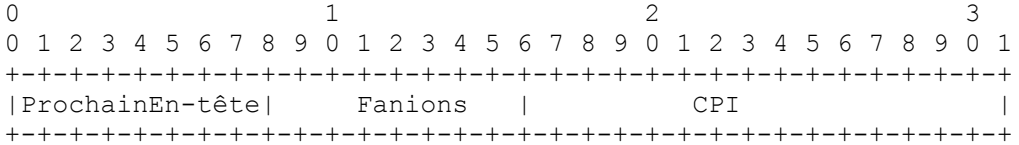
Prochain en-tête
Le champ Prochain en-tête est réglé à 108, Datagramme IPComp, [RFC1700].

Tous les autres champs d'en-tête IPv6 restent inchangés, y compris toutes les options d'en-tête non compressées.

L'en-tête IPComp est placé dans un paquet IPv6 en utilisant les mêmes règles que celles de l'en-tête de fragment IPv6. Cependant, si un paquet IPv6 contient à la fois un en-tête de fragment IPv6 et un en-tête IPComp, l'en-tête de fragment IPv6 DOIT précéder l'en-tête IPComp dans le paquet. Noter que d'autres en-têtes IPv6 peuvent être présents entre l'en-tête de fragment IPv6 et l'en-tête IPComp.

3.3 Structure d'en-tête IPComp

L'en-tête de quatre octets a la structure suivante :



Prochain en-tête
Sélecteur de 8 bits. Il mémorise le champ Protocole IPv4 ou le champ Prochain en-tête IPv6 de l'en-tête IP d'origine.

Fanions
Champ de 8 bits. Réservé pour utilisation future. Il DOIT être réglé à zéro. Il DOIT être ignoré par le nœud récepteur.

CPI *(Compression Parameter Index)* Indice de paramètre de compression
C'est un indice de 16 bits. Le CPI est mémorisé dans l'ordre des octets du réseau. Les valeurs de 0 à 63 désignent les algorithmes de compression bien connus, qui n'exigent pas d'informations supplémentaires, et sont utilisés pour l'établissement manuel. Les valeurs elles-mêmes sont identiques aux identifiants de transformation IPCOMP tels que définis dans la [RFC2407]. Consulter la [RFC2407] pour avoir l'ensemble initial de valeurs définies et les instructions sur la façon d'allouer de nouvelles valeurs. Les valeurs de 64 à 255 sont réservées pour les utilisations futures. Les valeurs de 256 à 61 439 sont négociées entre les deux nœuds pour la définition d'une association IPComp, comme défini à la section 4. Noter que lors de la négociation d'un des algorithmes bien connus, les nœuds PEUVENT choisir un CPI dans la gamme prédéfinie de 0 à 63. Les valeurs de 61 440 à 65 535 sont pour utilisation privée entre des parties mutuellement consentantes. Les deux nœuds participants peuvent choisir une valeur de CPI indépendamment l'une de l'autre et il n'y a pas de relation entre les deux CPI choisis séparément. L'en-tête IPComp sortant DOIT utiliser la valeur de CPI choisie par le nœud qui décompresse. Le CPI combiné avec l'adresse IP de destination identifie de façon univoque les caractéristiques de l'algorithme de compression pour le datagramme.

4. Négociation d'association IPComp (IPCA)

Pour utiliser le protocole IPComp, deux nœuds DOIVENT d'abord établir entre eux une association IPComp (IPCA). Une IPCA comporte toutes les information requises pour le fonctionnement de IPComp, y compris l'indice de paramètre de compression (CPI), le mode de fonctionnement, l'algorithme de compression à utiliser, et tout paramètre requis pour l'algorithme de compression choisi.

La politique d'établissement de IPComp peut être nœud par nœud et IPComp est alors appliqué à chaque paquet IP entre les nœuds, ou bien être une politique par session dans laquelle seules les sessions choisies utilisent IPComp entre les nœuds.

Deux nœuds peuvent choisir de négocier IPComp dans une direction ou dans les deux, et ils peuvent choisir d'employer un algorithme de compression différent dans chaque direction. Les nœuds DOIVENT cependant négocier un algorithme de compression dans chaque direction pour laquelle ils établissent une IPCA : il n'y a pas d'algorithme de compression par défaut.

Aucun algorithme de compression n'est obligatoire pour une mise en œuvre IPComp.

La IPCA est établie par des négociations dynamiques ou par configuration manuelle. Les négociations dynamiques DEVRAIENT utiliser le protocole d'échange de clés Internet (IKE) [RFC2409], lorsque IPsec est présent. Les négociations dynamiques PEUVENT être mises en œuvre par un protocole différent.

4.1 Utilisation de IKE

Pour IPComp dans le contexte de la sécurité IP, IKE fournit les mécanismes et lignes directrices nécessaires pour établir l'IPCA. En utilisant IKE, IPComp peut être négocié comme protocole autonome ou en conjonction avec d'autres protocoles IPsec.

Une association IPComp est négociée par l'initiateur en utilisant une proposition de charge utile, qui comporte une ou plusieurs charges utiles transformées. La proposition de charge utile spécifie le protocole de compression de charge utile IP dans le champ ID de protocole et chaque charge utile transformée contient le ou les algorithmes de compression spécifiques offerts à celui qui répond.

Le CPI est envoyé dans le champ SPI de la proposition avec le champ Taille de SPI réglé de façon à correspondre. Le CPI DEVRAIT être envoyé comme un nombre de 16 bits, avec le champ Taille de SPI réglé à 2. Autrement, le CPI PEUT être envoyé comme une valeur de 32 bit, avec le champ Taille SPI réglé à 4. Dans ce cas, le numéro de CPI de 16 bits DOIT être placé dans les deux octets de moindre poids du champ SPI, alors que les deux octets de poids fort DOIVENT être réglés à zéro, et DOIVENT être ignorés par le nœud qui reçoit. Le nœud qui reçoit DOIT être capable de traiter les deux formes de la proposition de CPI.

Dans le domaine d'interprétation (DOI, *Domain of Interpretation*) de la sécurité IP de l'Internet, IPComp est négocié

comme l'identifiant de protocole PROTO_IPCOMP. L'algorithme de compression est négocié comme un des identifiants de transformation IPCOMP défini.

Les attributs suivants sont applicables aux propositions IPComp :

Mode encapsulation
Pour proposer un mode encapsulation qui n'est pas par défaut (comme un mode tunnel) une proposition IPComp DOIT comporter un attribut Mode encapsulation. Si le mode encapsulation n'est pas spécifié, on suppose la valeur par défaut de mode Transport.

Durée de vie
Une proposition IPComp utilise les attributs Durée de vie et Type de vie pour suggérer une durée de vie à l'IPCA.

Lorsque IPComp est négocié au titre d'une suite de protection, toutes les offres qui sont logiquement reliées doivent être cohérentes. Cependant, une proposition IPComp NE DEVRAIT PAS inclure d'attributs qui ne sont pas applicables à IPComp. Une proposition IPComp NE DOIT PAS être rejetée parce qu'elle ne comporte pas dans la suite de protection d'attributs d'autres protocoles qui ne sont pas pertinents pour IPComp. Lorsque une proposition IPComp comporte de tels attributs, ceux-ci DOIVENT être ignorés lors de l'établissement de l'IPCA, et donc ignorés dans le fonctionnement de IPComp.

Note de mise en œuvre :
Un nœud peut éviter les calculs nécessaires pour déterminer l'algorithme de compression à partir du CPI si il utilise un des algorithmes bien connus ; cela peut gagner du temps dans le processus de décompression. Un nœud peut faire cela en négociant un CPI d'une valeur égale à l'identifiant de transformation prédéfini de cet algorithme de compression. Précisément, un nœud PEUT offrir un CPI dans la gamme prédéfinie en envoyant une proposition de charge utile qui DOIT contenir une seule charge utile transformée qui est identique au CPI. Lorsque il proposé deux charges utiles transformées ou plus, un nœud PEUT offrir des CPI dans la gamme prédéfinie en utilisant plusieurs propositions IPComp -- chacune DOIT comporter une seule charge utile transformée. Pour être clair : si une proposition de charge utile contient deux charges utiles transformées ou plus, le CPI DOIT être dans la gamme négociée. Un nœud receveur DOIT être capable de traiter chacune de ces formes proposées.

Note de mise en œuvre :
Les IPCA deviennent non uniques lorsque deux sessions IPComp, ou plus, sont établies entre deux nœuds, et que le même CPI bien connu est utilisé dans au moins deux des sessions. Les IPCA non uniques posent des problèmes pour le maintien des attributs spécifiques de chaque IPCA, qu'ils soient négociés (par exemple, la durée de vie) ou qu'ils soient internes (par exemple, les compteurs de l'algorithme adaptif pour traiter les charges utiles précédemment compressées). Pour s'assurer de l'unicité des IPCA entre deux nœuds, lorsque deux ou plus des IPCA utilisent le même algorithme de compression, les CPI DEVRAIENT être dans la gamme négociée. Cependant, Lorsque les IPCA ne sont pas obligés d'être uniques, par exemple, lorsque aucun attribut n'est utilisé pour ces IPCA, un CPI bien connu PEUT être utilisé. Pour être clair, lorsqu'il n'y a qu'une seule session d'établie entre deux nœuds qui utilisent un CPI bien connu particulier, cette IPCA est unique.

4.2 Utilisation de protocole non IKE

Les négociations dynamiques PEUVENT être mises en œuvre à travers un protocole autre que IKE. Un tel protocole sort du domaine d'application du présent document.

4.3 Configuration manuelle

Les nœuds peuvent établir des associations IPComp en utilisant une configuration manuelle. Pour cette méthode, un nombre limité d'indices de paramètres de compression (CPI, *Compression Parameters Indexes*) est désigné pour représenter une liste des méthodes de compression spécifiques.

5. Considérations pour la sécurité

Lorsque IPComp est utilisé dans le contexte de IPsec, on estime qu'il n'y a pas de conséquence sur les fonctions de sécurité sous-jacentes fournies par le protocole IPsec ; c'est-à-dire que l'utilisation de la compression n'est pas de nature à dégrader ou altérer l'architecture de sécurité sous-jacente ou les technologies de chiffrement utilisées pour le mettre en œuvre.

Lorsque IPComp est utilisé sans IPsec, la compression de charge utile IP peut réduire la sécurité de l'Internet, d'une façon similaire aux effets de l'encapsulation IP [RFC2003]. Par exemple, IPComp peut rendre difficile aux routeurs frontières de filtrer les datagrammes sur la base des champs d'en-tête. En particulier, la valeur d'origine du champ Protocole dans l'en-tête IP n'est pas située à ses positions normales au sein du datagramme, et les champs d'en-tête de couche transport au sein du datagramme, tels que les numéros d'accès, ne sont pas localisés dans leurs positions normales au sein du datagramme ni présentés dans leurs valeurs d'origine après la compression. Un routeur bordure de filtrage ne peut filtrer le datagramme que si il partage l'association IPComp utilisée pour la compression. Pour permettre cette sorte de compression dans des environnements dans lesquels tous les paquets doivent être filtrés (ou au moins décomptés) un mécanisme doit être en place pour que le nœud receveur communique en toute sécurité l'association IPComp au routeur bordure. Cela peut, plus rarement, s'appliquer aussi à l'association IPComp utilisée pour les datagrammes sortants.

6. Considérations relatives à l'IANA

Le présent document ne requiert aucune action de l'IANA. Les numéros bien connus utilisés dans ce document sont définis ailleurs ; voir la [RFC2407].

7. Changements depuis la RFC 2393

La présente section résume les changements apportés par ce document à la [RFC2393] et dont les mises en œuvre de la RFC 2393 devraient être informées. Tous les changements sont destinés à préciser la négociation d'une association IPComp (IPCA) en utilisant IKE [RFC2409] dans le contexte de IPsec.

1) Ajout de la précision que IPComp peut être négocié de façon autonome ou en liaison avec d'autres protocoles dans une suite de protection.

2) Définition du CPI dans le champ SPI d'une proposition IKE : le champ de deux octets est un DEVRAIT, le champ de quatre octets un PEUT. Définition du placement du CPI de 16 bits dans un champ de quatre octets. Spécifier qu'un receveur DOIT traiter les deux tailles de champ.

3) Ajout d'une phrase pour définir le mode Encapsulation par défaut comme mode Transport. Exigence qu'une proposition IPComp comporte un attribut Mode d'encapsulation lorsque il suggère une encapsulation qui n'est pas par défaut, telle qu'un mode Tunnel.

4) Ajout de l'attribut Durée de vie à la liste des attributs pris en charge (en plus de Mode de transport).

5) Spécification du traitement des attributs des transformations dans une suite de protection lorsque ils ne sont pas applicable à IPComp : ces attributs NE DEVRAIENT PAS être inclus dans une proposition IPComp et DOIVENT être ignorés lors de l'établissement d'une IPCA et dans le fonctionnement de IPComp. Les mises en œuvre de IPComp DOIVENT ne jamais rejeter une proposition IPCOMP qui ne comporte pas les attributs des autres transformations.

6) Ajout de notes de mise en œuvre sur la négociation et l'usage des CPI dans la gamme prédéfinie (bien connue).

8. Références

[RFC0791] J. Postel, éd., "Protocole Internet - Spécification du protocole du programme Internet", STD 5, septembre 1981.

[RFC1700] J. Reynolds et J. Postel, "Numéros alloués", STD 2, octobre 1994. *(Historique, voir www.iana.org))*

[RFC1962] D. Rand, "Protocole de contrôle de compression en PPP (CCP)", juin 1996.

[RFC2003] C. Perkins, "Encapsulation de IP dans IP", octobre 1996.

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

[RFC2393] A. Shacham et autres, "Protocole de compression de charge utile IP (IPComp)", décembre 1998. (*Obsolète, voir* RFC3173)

[RFC2407] D. Piper, "Le domaine d'interprétation de sécurité IP de l'Internet pour ISAKMP", novembre 1998. *(Obsolète, voir RFC4306)*

[RFC2409] D. Harkins et D. Carrel, "L'échange de clés Internet (IKE)", novembre 1998. *(Obsolète, voir la RFC4306)*

[RFC2460] S. Deering et R. Hinden, "Spécification du protocole Internet, version 6 (IPv6) ", décembre 1998. *(MàJ par la RFC5095, D.S)*

[V42BIS] Recommandation CCITT V.42 bis, "Procédures de compression de données pour équipement de terminaison de circuit de données (ETCD) utilisant les procédures de correction d'erreur", janvier 1990.

Adresse des auteurs

Abraham Shacham
Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
mél: shacham@shacham.net

Bob Monsour
18 Stout Road
Princeton, New Jersey 08540
USA
mél: bob@bobmonsour.com

Roy Pereira
Cisco Systems, Inc.
55 Metcalfe Street
Ottawa, Ontario K1P 6L5
Canada
mél : royp@cisco.com

Matt Thomas
3am Software Foundry
8053 Park Villa Circle
Cupertino, California 95014
USA
mél: matt@3am-software.com

Commentaires
Les commentaires devraient être addressés à ippcp@external.cisco.com mailing list et/ou aux auteurs.

Déclaration de droits de reproduction

Remerciement

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