Groupe de travail Réseau
Request for Comments : 3077
Catégorie : En cours de normalisation

Traduction Claude Brière de L'Isle

E. Duros, UDcast
W. Dabbous, INRIA Sophia-Antipolis
H. Izumiyama & N. Fujii, WIDE
Y. Zhang, HRL

mars 2001

Mécanisme de tunnelage de couche liaison pour liaisons unidirectionnelles

Résumé
Le présent document décrit un mécanisme pour émuler la pleine connexité bidirectionnelle entre tous les nœuds qui sont directement connectés par une liaison unidirectionnelle. Le "receveur" utilise un mécanisme de tunnelage de couche liaison pour transmettre les datagrammes aux "alimentations" sur un réseau bidirectionnel IP (*Internet Protocol*) séparé. Comme ils sont mis en œuvre à la couche de liaison, les protocoles qui s'ajoutent à IP peuvent aussi être pris en charge par ce mécanisme.

Table des Matières

1. Introduction

L'acheminement Internet et les protocoles de couche supérieure supposent que les liaisons sont bidirectionnelles, c'est-à-dire que des hôtes directement connectés peuvent communiquer l'un avec l'autre sur la même liaison.

Le présent document décrit un mécanisme de tunnelage de couche liaison qui permet à un ensemble de nœuds (alimentations et receveurs, voir la terminologie à la Section 2) qui sont directement connectés par une liaison unidirectionnelle pour envoyer des datagrammes comme si ils étaient tous connectés par une liaison bidirectionnelle. On présente une topologie générique à la Section 3 avec un mécanisme de tunnelage qui prend en charge plusieurs alimentations et receveurs. Noter que ce mécanisme n'est pas conçu pour les topologies où une paire de nœuds sont connectés par deux liaisons unidirectionnelles dans des directions opposées.

Le mécanisme de tunnelage exige que tous les nœuds aient une interface supplémentaire à une infrastructure interconnectée IP.

Le mécanisme de tunnelage est mis en œuvre à la couche liaison de l'interface de chaque nœud connecté à la liaison unidirectionnelle. Le but est de cacher aux couches supérieures, c'est-à-dire, à la couche réseau et au dessus, la nature unidirectionnelle de la liaison. Le mécanisme de tunnelage comporte aussi un protocole de configuration automatique de tunnel qui permet aux nœuds de s'activer/désactiver à tout moment.

L'encapsulation d'acheminement générique [RFC2784] est suggéré comme mécanisme de tunnelage car il donne un moyen pour porter IP, les datagrammes ARP, et tout autre protocole de couche 3 entre les nœuds.

Le mécanisme de tunnelage décrit dans le présent document a été discuté et accepté par le groupe de travail UDLR.

Dans le présent document, les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDÉ", "PEUT", et "FACULTATIF" sont à interpréter comme décrit dans le BCP 14, [RFC2119].

2. Terminologie

Liaison unidirectionnelle (UDL) : liaison de transmission à une seule direction, par exemple, une liaison de diffusion par satellite.

Receveur : routeur ou hôte qui a une connexité en réception seule avec une UDL.

Alimentation en envoi seul : routeur qui a une connexité en envoi seul avec une UDL.

Alimentation à capacité de réception : routeur qui a la connexité d'envoi et de réception avec une UDL.

Alimentation : alimentation en envoi seul ou à capacité de réception.

Nœud : receveur ou alimentation.

Interface bidirectionnelle : interface de communication normale qui peut envoyer ou recevoir des paquets, comme une carte Ethernet, un modem, etc.

3. Topologie

Les alimentations et les receveurs sont connectés via une liaison unidirectionnelle. Les alimentations en envoi seul peuvent seulement envoyer des données sur cette liaison unidirectionnelle, et les receveurs peuvent seulement recevoir d'elle des données. Les alimentations capables de réception ont les deux capacités d'envoi et de réception.

Ce mécanisme a été conçu pour fonctionner avec toute topologie avec un nombre quelconque de receveurs et une ou plusieurs alimentations. Cependant, on s'attend à ce que le nombre d'alimentations soit faible. En particulier, le cas spécial d'une seule alimentation en envoi seul et plusieurs receveurs fait partie des topologies prises en charge.

Un receveur a plusieurs interfaces, une interface en réception seule et une ou plusieurs interfaces supplémentaires de communication bidirectionnelles.

Une alimentation a plusieurs interfaces, une interface en envoi seul ou capable d'envoi et de réception connectée à la liaison unidirectionnelle et une ou plusieurs interfaces supplémentaires de communication bidirectionnelles. Une alimentation DOIT être un routeur.

Les tunnels sont construits entre les interfaces bidirectionnelles des nœuds, de sorte que ces interfaces doivent être interconnectées par une infrastructure IP. Dans le présent document, on suppose que cette infrastructure est l'Internet.

La Figure 1 décrit une topologie générique avec plusieurs alimentations et plusieurs receveurs.

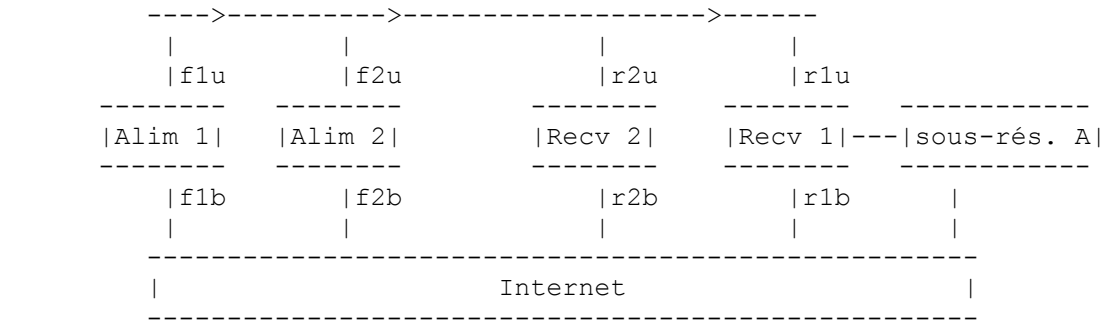


Figure 1 : Topologie générique

f1u (respectivement f2u) est l'adresse IP de l'interface 'Alim 1' (resp. Alim 2) en envoi seul.

f1b (resp. f2b) est l'adresse IP de l'interface 'Alim 1' (resp. Alim 2) bidirectionnelle connectée à l'Internet.

r1u (resp. r2u) est l'adresse IP de l'interface 'Receveur 1' (resp. Receveur 2) en réception seule.

r1b (resp. r2b) est l'adresse IP de l'interface 'Receveur 1' (resp. Receveur 2) bidirectionnelle connectée à l'Internet.

Sous-réseau A est un réseau de zone locale connecté à recv1.

Noter que les nœuds ont des adresses IP sur leurs interfaces unidirectionnelles et bidirectionnelles. Les adresses sur les interfaces unidirectionnelles (f1u, f2u, r1u, r2u) seront tirées du même réseau IP. En général les adresses sur les interfaces bidirectionnelles (f1b, f2b, r1b, r2b) seront tirées de réseaux IP différents, et l'Internet fera l'acheminement entre elles.

4. Problèmes des liaisons unidirectionnelles

Les interfaces en réception seule sont "muettes" et les interfaces en envoi seul sont "sourdes". Donc, un datagramme passé au pilote de couche liaison d'une interface en réception seule est simplement éliminé. La couche liaison d'une interface en envoi seul ne reçoit jamais rien.

La couche réseau n'a pas connaissance de la technologie de transmission sous-jacente sauf qu'il considère son accès comme bidirectionnel. Fondamentalement, pour les datagrammes sortants, la couche réseau choisit le premier bond correct sur le réseau connecté selon un tableau d'acheminement et passe le ou les paquets au pilote de couche liaison approprié.

En se référant à la Figure 1, Recv 1 et Alim 1 appartiennent au même réseau. Cependant, si Recv 1 initie un 'ping f1u', il ne peut pas obtenir une réponse de Alim 1. La couche réseau de Recv 1 livre le paquet au pilote de l'interface en réception seule, qui évidemment ne peut pas l'envoyer à l'alimentation.

De nombreux protocoles de l'Internet supposent que les liaisons sont bidirectionnelles. En particulier, les protocoles d'acheminement utilisés par des routeurs directement connectés ne se comportent plus correctement en présence d'une liaison unidirectionnelle.

5. Émulation d'un réseau de diffusion bidirectionnel

La plus simple solution est d'émuler un réseau de couche liaison capable de diffusion. Cela va permettre un déploiement immédiat sans changement des protocoles de niveau supérieur existants Bien que d'autres structures de réseau, telles que NBMA, puissent aussi être émulées, un réseau de diffusion est plus généralement utile. Bien qu'un réseau de couche 3 puisse être émulé, un réseau de couche liaison permet l'utilisation immédiate de tout autre protocole de couche réseau, et tout particulièrement permet l'utilisation immédiate de ARP.

Un mécanisme de tunnelage de couche liaison qui émule la connexité bidirectionnelle en présence d'une liaison unidirectionnelle sera décrit à la Section suivante. On considère d'abord les divers scénarios de communication qui caractérisent un réseau de diffusion afin de définir quelles fonctionnalités doit effectuer le mécanisme de tunnelage de couche liaison afin d'émuler une liaison de diffusion bidirectionnelle.

On énumère ici les scénarios qui seraient faisables sur un réseau de diffusion, c'est-à-dire, si les alimentations et les receveurs étaient connectés par une liaison de diffusion bidirectionnelle :

Scénario 1 : Un receveur peut envoyer un paquet à une alimentation (communication point à point entre un receveur et une alimentation).

Scénario 2 : Un receveur peut envoyer un paquet en diffusion/diffusion groupée sur la liaison à tous les nœuds (point à multipoint).

Scénario 3 : Un receveur peut envoyer un paquet à un autre receveur (communication point à point entre deux receveurs).

Scénario 4 : Une alimentation peut envoyer un paquet à une alimentation en envoi seul (communication point à point entre deux alimentations).

Scénario 5 : Une alimentation peut envoyer un paquet en diffusion/diffusion groupée sur la liaison à tous les nœuds (point à multipoint).

Scénario 6 : Une alimentation peut envoyer un paquet à un receveur ou à une alimentation capable de recevoir (point à point).

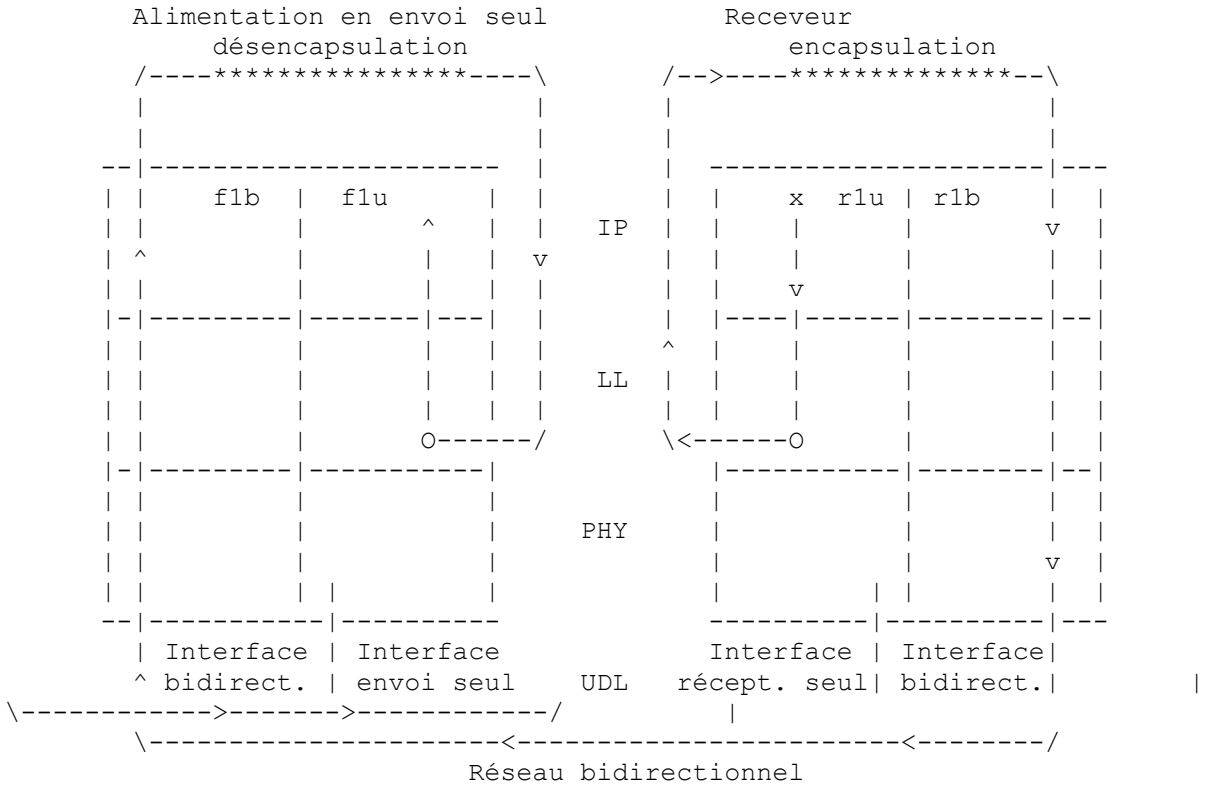
Ces scénarios sont possibles sur un réseau de diffusion. Le scénario 6 est déjà faisable sur la liaison unidirectionnelle. Le mécanisme de tunnelage de couche liaison devrait donc fournir la fonctionnalité de prise en charge des scénarios 1 à 5.

Noter que la transmission IP normale sur une telle émulation de réseau (c'est-à-dire, utilisant le réseau émulé comme un réseau de transit) fonctionne correctement ; l'adresse de prochain bond chez le receveur sera l'adresse de la liaison unidirectionnelle d'un autre routeur (une alimentation ou un receveur) qui va alors relayer le paquet.

6. Mécanisme de tunnelage de couche liaison

Ce mécanisme de tunnelage de couche de liaison fonctionne en dessous de la couche réseau. Il vise à émuler la connexité de couche liaison. C'est transparent pour la couche réseau : la liaison apparaît et se comporte à la couche réseau comme si elle était bidirectionnelle.

La Figure 2 décrit une représentation en couches du mécanisme de tunnelage de couche de liaison dans le cas du scénario 1.



x : la couche IP chez le receveur génère un datagramme à transmettre sur l'interface en réception seule.
O : Point d'entrée où le mécanisme de tunnelage de couche liaison est déclanché.

Figure 2 : Scénario 1 utilisant le mécanisme de tunnelage de couche liaison

6.1 Mécanisme de tunnelage chez le receveur

Chez le receveur, un datagramme est livré à la couche de liaison de l'interface unidirectionnelle pour transmission (voir la Figure 2). Il est alors encapsulé au sein d'un en-tête de commande d'accès au support physique (MAC, *Media Access Control*) correspondant à la liaison unidirectionnelle. Ce paquet ne peut pas être envoyé directement sur la liaison, de sorte qu'il est alors traité par le mécanisme de tunnelage.

Le paquet est encapsulé au sein d'un en-tête IP dont la destination est l'adresse IP d'une interface d'alimentation bidirectionnelle (f1b ou f2b). Cette adresse de destination est aussi appelée le point d'extrémité de tunnel. Le mécanisme pour qu'un receveur apprenne ces adresses et choisisse l'alimentation est expliqué à la Section 7. Le type d'encapsulation est décrit à la Section 8.

Dans tous les cas, le paquet est encapsulé, mais le point d'extrémité de tunnel (et l'adresse IP) dépend de l'adresse MAC de destination du paquet encapsulé. Si l'adresse MAC de destination est :

1) l'adresse MAC d'une interface d'alimentation connectée à la liaison unidirectionnelle (scénario 1). Le datagramme est encapsulé, l'adresse de destination du datagramme encapsulant est le point d'extrémité du tunnel d'alimentation (f1b en se référant à la Figure 2).

2) une adresse MAC de diffusion/diffusion groupée (scénario 2). Le datagramme est encapsulé, l'adresse de destination du datagramme encapsulant est le point d'extrémité du tunnel d'alimentation par défaut. Voir au paragraphe 7.4 les détails sur l'alimentation par défaut.

3) une adresse MAC qui appartient au réseau unidirectionnel mais n'est pas une adresse d'alimentation (scénario 3). Le datagramme est encapsulé, l'adresse de destination du datagramme encapsulant est le point d'extrémité du tunnel d'alimentation par défaut.

Le datagramme encapsulé est passé à la couche réseau qui le transmet selon son adresse de destination. L'adresse de destination est une interface bidirectionnelle d'alimentation qui est accessible via l'Internet. Dans ce cas, le datagramme encapsulé est transmis via l'interface bidirectionnelle du receveur (r1b).

6.2. Mécanisme de tunnelage chez l'alimenteur

Une alimentation traite les paquets en rapport avec la liaison unidirectionnelle de deux façons différentes :
- les paquets générés par une application locale ou les paquets acheminés comme d'habitude par la couche IP peuvent devoir être transmis sur la liaison unidirectionnelle (paragraphe 6.2.1) ;
- les paquets encapsulés reçus d'un autre receveur ou alimentation ont besoin d'un traitement de tunnel (paragraphe 6.2.2).

Une alimentation ne peut pas envoyer directement un paquet à une alimentation en envoi seul sur la liaison unidirectionnelle (scénario 4). Afin d'émuler ce type de communication, les alimentations doivent tunneler les paquets aux alimentations en envoi seul. Une alimentation DOIT tenir une liste de tous les autres points d'extrémité de tunnel d'alimentation. Cette liste DOIT indiquer quels sont les points d'extrémité de tunnel d'alimentation en envoi seul. Ceci est configuré manuellement chez l'alimentation par l'administrateur local, comme décrit à la Section 7.

6.2.1 Transmission des paquets sur la liaison unidirectionnelles

Lorsque un datagramme est livré à la couche de liaison de l'interface unidirectionnelle d'une alimentation pour transmission, son traitement dépend de l'adresse MAC de destination du paquet. Si l'adresse MAC de destination est :
1) l'adresse MAC d'un receveur ou d'une alimentation capable de réception (scénario 6) ; le paquet est envoyé sur la liaison unidirectionnelle. C'est la "transmission" classique.
2) l'adresse MAC d'une alimentation en envoi seul (scénario 4). Le paquet est encapsulé et envoyé au point d'extrémité du tunnel d'alimentation en envoi seul. Le type d'encapsulation est décrit à la Section 8.
3) une destination de diffusion/diffusion groupée (scénario 5). Le paquet est envoyé sur la liaison unidirectionnelle. En même temps, une copie de ce paquet est encapsulée et envoyée à chaque alimentation de la liste des points d'extrémité du tunnel d'alimentation en envoi seul. Ainsi, la diffusion/diffusion groupée va atteindre tous les receveurs et toutes les alimentations en envoi seul.

6.2.2 Réception des paquets encapsulés

Les alimentations écoutent les datagrammes encapsulés entrant sur leurs points d'extrémité de tunnel. Les paquets encapsulés auront été reçus sur une interface bidirectionnelle, et auront parcouru leur chemin en remontant la pile de protocoles IP. Ils vont ensuite entrer dans un processus de désencapsulation (voir la Figure 2).

La désencapsulation révèle le paquet de couche de liaison original. Noter que ceci n'a en aucune façon été modifié par les routeurs intermédiaires ; en particulier, l'en-tête MAC d'origine sera intact.

La suite des actions dépend de l'adresse MAC de destination du paquet de couche de liaison, qui peut être :
1) L'adresse MAC de l'interface d'alimentation connectée à la liaison unidirectionnelle, c'est-à-dire, sa propre adresse MAC (scénarios 1 et 4). Le paquet est passé à la couche de liaison de l'interface connectée à la liaison unidirectionnelle qui peut alors le livrer aux couches supérieures. Il en résulte que le datagramme est traité comme si il provenait de la liaison unidirectionnelle, et était livré en local. Les scénarios 1 et 4 sont maintenant faisables, un receveur ou une alimentation peut envoyer un paquet à une alimentation.
2) Une adresse de receveur (scénario 3). Le paquet est passé à la couche de liaison de l'interface connectée à la liaison unidirectionnelle. Il est directement envoyé sur la liaison unidirectionnelle, au receveur indiqué. Noter que le paquet ne doit pas être livré en local. Le scénario 3 est maintenant faisable, un receveur peut envoyer un paquet à un autre receveur.
3) une adresse de diffusion/diffusion groupée, cela correspond aux scénarios 2 et 5. On doit distinguer deux cas, soit (i) le paquet encapsulé a été envoyé d'un receveur, soit (ii) le paquet encapsulé a été envoyé d'une alimentation (paquet encapsulé en diffusion/diffusion groupée envoyé à une alimentation en envoi seul). Ces cas sont distingués en examinant l'adresse de source du paquet encapsulant et en le comparant avec la liste des adresses IP d'alimentation configurées. L'action prise alors est :
   i) l'alimentation a été conçue comme une alimentation par défaut par un receveur pour transmettre le paquet en diffusion/diffusion groupée. L'alimentation est alors chargée d'envoyer le paquet en diffusion groupée à tous les nœuds. La livraison à tous les nœuds est accomplie en exécutant les trois actions suivantes :
      - le paquet est encapsulé et envoyé à la liste des points d'extrémité de tunnel d'alimentation en envoi seul ;
      - aussi, le paquet est passé à la couche de liaison de l'interface qui le transmet directement sur la liaison unidirectionnelle (tous les receveurs et alimentations capables de le recevoir) ;
      - et aussi, la couche de liaison le livre en local aux couches supérieures.

      Attention : un receveur qui envoie un paquet encapsulé en diffusion/diffusion groupée à une alimentation par défaut va recevoir son propre paquet via la liaison unidirectionnelle. Un filtrage correct comme décrit dans la [RFC1112] doit être appliqué.
   ii) L'alimentation reçoit le paquet et le garde pour livraison locale. Le paquet est passé à la couche de liaison de

l'interface connectée à la liaison unidirectionnelle qui le livre aux couches supérieures.

Le scénario 2 est maintenant faisable, un receveur peut envoyer un paquet en diffusion/diffusion groupée sur la liaison unidirectionnelle et il sera entendu par tous les nœuds.

7. Protocole de configuration dynamique de tunnel (DTCP)

Les receveurs et alimentations ont à connaître les points d'extrémité de tunnel d'alimentation afin de transmettre les datagrammes encapsulés (par exemple, scénarios 1 et 4).

Le nombre des alimentations est supposé être relativement petit (Section 3) de sorte que à chaque alimentation, la liste de toutes les alimentations est configurée manuellement. Cette liste devrait noter quelles sont les alimentations en envoi seul, et quelles sont les alimentations capables de recevoir. L'administrateur établit les tunnels avec toutes les alimentations en envoi seul. Un point d'extrémité de tunnel est une adresse IP d'une liaison bidirectionnelle sur une alimentation en envoi seul.

Pour des raisons d'adaptabilité, la configuration manuelle ne peut pas être faite chez les receveurs. Les tunnels doivent être configurés et entretenus de façon dynamique par les receveurs, pour l'adaptabilité mais aussi pour faire face aux événements suivants :

1) Nouvelle détection d'alimentation. Lorsque une nouvelle alimentation est activée, chaque receveur doit créer un tunnel pour activer la communication bidirectionnelle avec elle.
2) Perte de la détection de liaison unidirectionnelle. Lorsque une liaison unidirectionnelle est désactivée, les receveurs doivent désactiver leurs tunnels. Le mécanisme de tunnelage émule la connexité bidirectionnelle entre les nœuds. Donc, si la liaison unidirectionnelle est désactivée, une alimentation ne devrait pas recevoir de datagrammes des receveurs. Les protocoles qui considèrent une liaison comme opérationnelle si ils en reçoivent des datagrammes (par exemple, le protocole RIP [RFC2453]) exigent ce comportement pour un fonctionnement correct.
3) Perte de la détection d'alimentation.. Lorsque une alimentation est désactivée, les receveurs doivent désactiver leurs tunnels correspondants. Cela empêche que des datagrammes soient inutilement tunnelés, ce qui pourrait surcharger l'Internet. Par exemple, il n'est pas besoin que les receveurs transmettent un message en diffusion à travers un tunnel dont le point d'extrémité est fermé.

Le protocole DTCP fournit un moyen pour que les receveurs découvrent de façon dynamique la présence d'alimentations et tiennent une liste des points d'extrémité de tunnel opérationnels. Les alimentations annoncent périodiquement leurs adresses de point d'extrémité de tunnel sur la liaison unidirectionnelle. Les receveurs écoutent des annonces et tiennent une liste des points d'extrémité de tunnel.

7.1 Message HELLO

Le protocole DTCP est un protocole 'unidirectionnel', les messages ne sont envoyés que des alimentations aux receveurs.

Le format de paquet est montré à la Figure 3. Les champs contiennent des entiers binaires, dans l'ordre normal de l'Internet avec le bit de poids fort en premier. Chaque marque représente un bit.

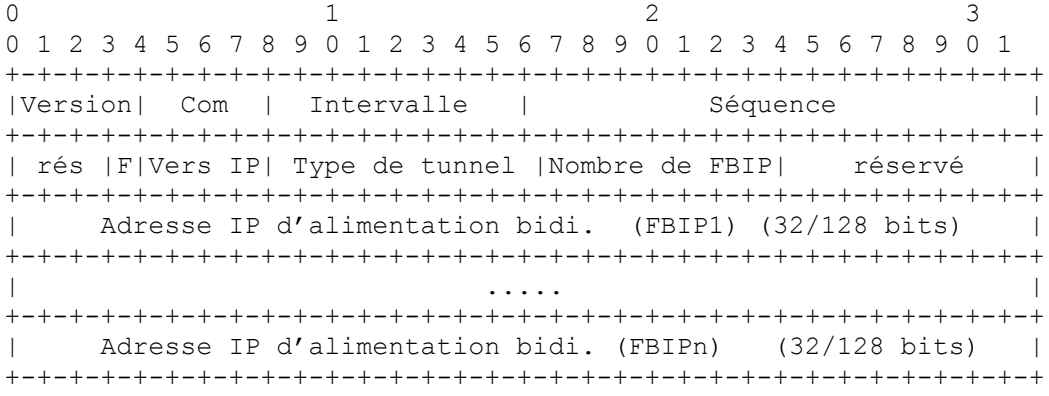


Figure 3 : Format de paquet

Chaque datagramme contient les champs suivants ; noter que les constantes sont écrites en majuscules et sont définies au

paragraphe 7.5 :

Version (entier non signé de 4 bits) : numéro de version DTCP. DOIT être DTCP_VERSION.

Com (entier non signé de 4 bits) : Champ Commande ; les valeurs possibles sont :
1 - JOINDRE : message annonçant que l'alimentation qui envoie ce message est active et fonctionne.
2 - QUITTER : message annonçant que l'alimentation qui envoie ce message est en train de fermer.

Intervalle (entier non signé de 8 bits) : Intervalle en secondes entre les messages HELLO pour le protocole IP dans "version IP". Doit être > 0. La valeur recommandée est HELLO_INTERVAL. Si cette valeur est augmentée, l'alimentation DOIT continuer d'envoyer des messages HELLO à l'ancien taux pendant au moins l'ancienne période HELLO_LEAVE.

Séquence (entier non signé de 16 bits) : valeur aléatoire initialisée au moment de l'amorçage et incrémentée de 1 chaque fois qu'une valeur de message HELLO est modifiée.

rés (3 bits) : champ réservé/non utilisé ; DOIT être à zéro.

F (1 bit) : bit indiquant le type d'alimentation :
0 = alimentation en envoi seul
1 = alimentation capable de réception

Vers IP (entier non signé de 4 bits) : version du protocole IP des adresses IP de l'alimentation bidirectionnelle (FBIP, *Feed Bidirectional interface IP address*) :
4 = IP version 4
6 = IP version 6

Type de tunnel (entier non signé de 8 bits) : protocole de tunnelage pris en charge par l'alimentation. Cette valeur est le numéro de protocole IP défini dans la [RFC1700] et ses successeurs légitimes. Les receveurs DOIVENT utiliser cette forme d'encapsulation de tunnel lors d'un tunnelage à l'alimentation.
47 = GRE [RFC2784] (recommandé)
D'autres types de protocoles permettant l'encapsulation de couche de liaison sont permis. L'obtention de nouvelles valeurs est documenté dans la [RFC2780].

Nombre de FBIP (entier non signé de 8 bits) : nombre d'adresses IP d'alimentation bidirectionnelle qui sont énumérées dans le message HELLO réservé de (8 bits) : Champ réservé/non utilisé, DOIT être zéro.

Adresse IP d'alimentation bidi. (32 ou 128 bits) : c'est l'adresse IP d'une interface bidirectionnelle d'alimentation (point d'extrémité de tunnel) accessible via l'Internet. Une alimentation a 'Nombre de FBIP' adresses IP qui sont des points d'extrémité de tunnel opérationnels. Ils sont énumérés dans l'ordre de préférence, FBIP1 étant le point d'extrémité de tunnel le plus convenable.

7.2 DTCP sur l'alimentation : envoi des paquets HELLO

Le protocole DTCP fonctionne par dessus UDP. Les paquets sont envoyés à l'adresse de diffusion groupée "annonce DTCP" sur la liaison unidirectionnelle à l'accès HELLO_PORT avec un TTL de 1. À cause des déploiements existants, une alimentation DEVRAIT aussi prendre en charge l'utilisation de l'ancienne adresse d'annonce DTCP, comme décrit dans l'Appendice B.

L'adresse source du paquet HELLO est réglée à l'adresse IP de l'interface d'alimentation connectée à la liaison unidirectionnelle. Dans le reste de ce document, cette valeur est appelée adresse IP d'alimentation unidirectionnelle (FUIP, *Feed Unidirectional IP address*).

Le processus chargé de l'envoi des paquets HELLO remplit chaque champ du datagramme conformément à la description du paragraphe 7.1.

Tant qu'une alimentation est active et en fonctionnement, elle annonce périodiquement sa présence aux receveurs. Elle DOIT envoyer des paquets HELLO contenant une commande JOIN tous les HELLO_INTERVAL sur la liaison unidirectionnelle.

En se référant à la Figure 1 de la Section 3, Alim 1 (resp. Alim 2) envoie des messages HELLO avec le champ FBIP1 réglé à f1b (resp. f2b).

Lorsque l'alimentation est sur le point de fermer, ou lorsque l'acheminement sur la liaison unidirectionnelle est sur le point d'être intentionnellement interrompu, il est recommandé que les alimentations :
1) arrêtent d'envoyer des messages HELLO contenant une commande JOIN,
2) envoient un message HELLO contenant une commande LEAVE pour informer les receveurs que l'alimentation n'effectue plus d'acheminement sur la liaison unidirectionnelle.

7.3 DTCP chez le receveur : réception des paquets HELLO

Sur la base de la réception de messages HELLO, les receveurs découvrent la présence d'alimentations, tiennent une liste des alimentations actives, et gardent trace des points d'extrémité de tunnel pour ces alimentations.

Pour chaque alimentation active, et pour chaque protocole IP pris en charge, au moins les informations suivantes seront conservées :

FUIP	- adresse IP d'alimentation de liaison unidirectionnelle
FUMAC	- adresse MAC correspondant à l'adresse IP ci-dessus
(FBIP1,...,FBIPn)	- liste des points d'extrémité de tunnel
type de tunnel	- type de tunnel pris en charge par cette alimentation
Séquence	- valeur de "séquence" du dernier HELLO reçu de cette alimentation
temporisateur	- utilisé pour limiter la durée de vie de cette entrée

La valeur FUMAC pour une alimentation active est nécessaire pour le fonctionnement de ce protocole. Cependant, la méthode de découverte de cette valeur n'est pas spécifiée ici.

Initialement, la liste des alimentations actives est vide.

Lorsque un receveur démarre, il DOIT faire fonctionner un processus qui se joint au groupe de diffusion groupée "annonce DTCP" et écoute les paquets entrants sur l'accès HELLO_PORT provenant de la liaison unidirectionnelle.

À réception d'un message HELLO, le processus vérifie le numéro de version du protocole. Si il est différent de HELLO_VERSION, le paquet est éliminé et le processus attend le prochain paquet entrant.

Après vérification réussie du numéro de version, les actions ultérieures dépendent du type de commande :

- JOIN :

Le processus vérifie si l'adresse FUIP appartient déjà à la liste des alimentations actives.
Si elle n'y est pas, il crée une nouvelle entrée, pour la FUIP d'alimentation, et l'ajoute à la liste des alimentations actives. Le nombre d'adresses IP d'alimentation bidirectionnelle à lire est déduit du champ 'Nombre de FBID'. Ces points d'extrémité de tunnel (FBIP1,...,FBIPn) peuvent alors être ajoutés à la nouvelle entrée. Les valeurs de type de tunnel et de séquence sont aussi prises dans le paquet HELLO et enregistrées dans la nouvelle entrée. Un temporisateur réglé à HELLO_LEAVE est associé à cette entrée.
Si elle y est, le numéro de séquence est comparé au numéro de séquence contenu dans le paquet tHELLO précédent envoyé par cette alimentation. Si ils sont égaux, le temporisateur associé à cette entrée est remis à HELLO_LEAVE. Autrement, toutes les informations correspondant à la FUIP sont réglées aux valeurs tirées du paquet HELLO.
En se référant à la Figure 1 de la Section 3, les receveurs (recv 1 et recv 2) ont une liste des alimentations actives qui contient deux entrées : Alim 1 avec une FUIP de f1u et une liste des points d'extrémité de tunnel (f1b) ; et Alim 2 avec une FUIP de f2u et une liste des points d'extrémité de tunnel (f2b).

- LEAVE :

Le processus vérifie si il y a une entrée pour la FUIP dans la liste des alimentations actives. Si il y en a une, le temporisateur est désactivé et l'entrée est supprimée de la liste. Le message LEAVE donne un moyen de mettre rapidement à jour la liste des alimentations actives.

Une fin de temporisation survient pour l'une des deux raisons suivantes :
1) une alimentation a fermé sans envoyer de message LEAVE. Comme les messages JOIN ne sont plus envoyés de cette alimentation, une fin de temporisation se produit HELLO_LEAVE après le dernier message JOIN.
2) la liaison unidirectionnelle est fermée. Donc aucun autre message JOIN n'est reçu des alimentations, et elles vont toutes arriver indépendamment en fin de temporisation. La fin de temporisation de chaque entrée dépend de sa valeur individuelle de HELLO_LEAVE, et de quand le dernier message JOIN a été envoyé par cette alimentation, avant la fermeture de la liaison unidirectionnelle.

Dans l'un et l'autre cas, la connexité bidirectionnelle ne peut plus être assurée entre le receveur et l'alimentation (FUIP) : soit l'alimentation n'achemine plus de datagrammes sur la liaison unidirectionnelle, soit la liaison est fermée. Donc, l'entrée associée est retirée de la liste des alimentations actives, quelle qu'en soit la cause. Il en résulte que la liste ne contient que des points d'extrémité de tunnel opérationnels.

Le protocole HELLO fournit aux receveurs une liste des alimentations, et une liste des points d'extrémité de tunnel utilisables (FBIP1,..., FBIPn) pour chaque alimentation. Dans le paragraphe qui suit, on décrit comment intégrer le protocole HELLO dans le mécanisme de tunnelage décrit aux paragraphes 6.1 et 6.2.

7.4 Mécanisme de tunnelage en utilisant la liste des alimentations actives

Ce paragraphe explique comment le mécanisme de tunnelage utilise la liste des alimentations actives pour traiter les datagrammes qui sont à tunneler. En se référant au paragraphe 6.1, il montre comment sont choisis les points d'extrémité de tunnel d'alimentation.

Le choix de l'alimentation par défaut est fait de façon indépendante à chaque receveur. Le choix est une affaire de politique locale, et cette politique sort du domaine d'application du présent document. Cependant, à titre d'exemple, l'alimentation par défaut peut être l'alimentation qui a le plus faible délai d'aller retour avec le receveur.

Lorsque un receveur envoie un paquet à une alimentation, il doit choisir un point d'extrémité de tunnel sur la liste des FBIP. La 'FBIP préférée' est généralement FBIP1 (paragraphe 7.1). Pour diverses raisons, un receveur peut décider d'utiliser une FBIP différente, disons FBIPi au lieu de FBIP1, comme point d'extrémité de tunnel. Par exemple, le receveur peut avoir une meilleure connexité à FBIPi. Cette décision est prise par l'administrateur du receveur.

On montre ici comment la liste des alimentations actives est impliquée lorsque un receveur tunnelle un paquet de couche de liaison. Le paragraphe 6.1 faisait la liste des cas suivants, selon que l'adresse MAC de destination du paquet est :

1) l'adresse MAC d'une interface d'alimentation connectée à la liaison unidirectionnelle : ceci est VRAI si l'adresse correspond à une adresse FUMAC dans la liste des alimentations actives. Le paquet est tunnelé à la FBIP préférée de l'alimentation correspondante ;
2) l'adresse de diffusion de la liaison unidirectionnelle ou une adresse de diffusion groupée : ceci est déterminé par les règles de format d'adresse MAC, et la liste des alimentations actives n'est pas impliquée. Le paquet est tunnelé à la FBIP préférée de l'alimentation par défaut ;
3) une adresse qui appartient au réseau unidirectionnel mais n'est pas une adresse d'alimentation : ceci est VRAI si l'adresse n'est ni de diffusion ni de diffusion groupée, ni trouvée dans la liste des alimentations actives. Le paquet est tunnelé à la FBIP préférée de l'alimentation par défaut.

Dans tous les cas, le type d'encapsulation dépend du type de tunnel exigé par l'alimentation qui est choisie.

7.5 Définitions des constantes

DTCP_VERSION est 1.
HELLO_INTERVAL est 5 secondes.
Le groupe de diffusion groupée "annonce DTCP" est 224.0.0.36, alloué par l'IANA.
HELLO_PORT est 652. C'est un accès système réservé alloué par l'IANA, aucun autre trafic ne doit être permis.
HELLO_LEAVE est 3*Intervalle, comme annoncé dans un paquet HELLO, c'est-à-dire, 15 secondes si le HELLO_INTERVAL par défaut a été annoncé.

8. Format d'encapsulation de tunnel

Le mécanisme de tunnelage fonctionne à la couche de liaison et émule la connexité bidirectionnelle entre les receveurs et les alimentations. On suppose que le matériel connecté à la liaison unidirectionnelle prend en charge l'adressage MAC en diffusion et en envoi individuel. C'est à dire que l'alimentation peut envoyer un paquet à un receveur particulier en utilisant l'adresse de destination MAC en envoi individuel ou à un en semble de receveurs en utilisant une adresse de destination en diffusion/diffusion groupée. Le matériel (ou le pilote) du receveur peut alors filtrer les paquets entrants envoyés sur les liaisons unidirectionnelles sans faire aucune hypothèse sur le type de données encapsulées.

D'une façon similaire, un receveur devrait être capable d'envoyer des paquets MAC en envoi individuel et en diffusion via ses tunnels. Les paquets de couche liaison sont encapsulés. Il en résulte qu'après la désencapsulation d'un paquet entrant,

l'alimentation peut effectuer un filtrage de couche de liaison comme si les données venaient directement de la liaison unidirectionnelle (voir la Figure 2).

L'encapsulation d'acheminement générique (GRE, *Generic Routing Encapsulation*) [RFC2784] convient à nos exigences parce qu'elle spécifie un protocole pour encapsuler des paquets arbitraires, et permet d'utiliser IP comme protocole de livraison.

L'administrateur local de l'alimentation décide quelle encapsulation il va demander qu'utilisent les receveurs, et règle le champs Type de tunnel dans le message HELLO en conséquence. La valeur 47 (décimal) indique GRE. D'autres valeurs peuvent être utilisées, mais leur interprétation doit faire l'objet d'un accord entre les alimentations et les receveurs. Un tel usage n'est pas défini ici.

8.1 Encapsulation d'acheminement générique chez le receveur

Un paquet GRE se compose d'un en-tête dans lequel un champ Type spécifie le protocole encapsulé (ARP, IP, IPX, etc.). Voir dans la [RFC2784] les détails sur l'encapsulation. Dans notre cas, seule la prise en charge du schéma d'adressage MAC de la liaison unidirectionnelle DOIT être mise en œuvre.

Un paquet tunnelé avec l'encapsulation GRE a le format suivant : l'en-tête de livraison est un en-tête IP dont la destination est le point d'extrémité du tunnel (FBIP), suivi par un en-tête GRE qui spécifie le type de couche de liaison de la liaison unidirectionnelle. La Figure 4 présente le paquet encapsulé entier.

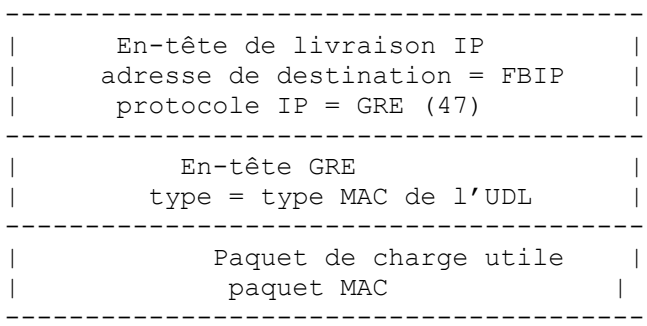


Figure 4 : Paquet encapsulé

9. Problèmes

9.1 Résolution d'adresse de matériel

Sans considérer si la liaison est unidirectionnelle ou bidirectionnelle, si une alimentation envoie un paquet sur un type de réseau non point à point, elle exige l'adresse de liaison de données de la destination. ARP [RFC826] est utilisé sur les réseaux Ethernet à cette fin.

Le mécanisme de couche de liaison émule un réseau bidirectionnel en présence d'une liaison unidirectionnelle. Cependant, il y a des délais asymétriques entre chaque paire (alimentation, receveur). Le canal de retour entre un receveur et une alimentation a des délais variables parce que les paquets passent à travers l'Internet. De plus, un exemple typique de liaison unidirectionnelle est une liaison satellite GEO dont le délai est d'environ 250 millisecondes.

À cause des longs délais d'aller-retour, les méthodes de résolution d'adresse réactives comme ARP [RFC826] peuvent ne pas fonctionner bien. Par exemple, une alimentation peut avoir à transmettre des paquets à des débits de données élevés à un receveur dont l'adresse de matériel est inconnue. Le flux de paquets est passé au pilote de la couche de liaison de l'interface en envoi seul de l'alimentation. Lorsque le premier paquet arrive, la couche de liaison réalise qu'elle n'a pas l'adresse de matériel correspondant au prochain bond, et envoie une demande ARP. Alors que la couche de liaison attend la réponse (au moins 250 ms pour le cas du satellite GEO) les paquets IP sont mis en mémoire tampon par l'alimentation. Si elle se trouve à court d'espace avant que la réponse ARP arrive, les paquets IP seront éliminés.

Ce problème de protocoles de résolution d'adresse n'est pas traité dans le présent document. Une solution ad hoc est possible lorsque l'adresse MAC est configurable, ce qui est possible dans certaines cartes de receveur de satellite. Une transformation simple (peut-être nulle) de l'adresse IP peut être utilisée comme adresse MAC. Dans ce cas, les envoyeurs

n'ont pas besoin de "résoudre" une adresse IP en adresse MAC, ils ont juste besoin d'effectuer la simple transformation.

9.2 Protocoles d'acheminement

Le mécanisme de tunnelage de couche de liaison cache au réseau et aux couches supérieures le fait que les alimentations et les receveurs sont connectés par une liaison unidirectionnelle. La communication est bidirectionnelle, mais asymétrique en bande passante et en délais.

Afin d'incorporer les liaisons unidirectionnelles dans l'Internet, les alimentations et les receveurs peuvent devoir faire fonctionner des protocoles d'acheminement dans certaines topologies. Ces protocoles vont bien fonctionner parce que le mécanisme de tunnelage résulte en une connexité bidirectionnelle entre toutes les alimentations et receveurs. Les messages d'acheminement peuvent donc être échangés comme sur n'importe quel réseau bidirectionnel.

Le mécanisme de tunnelage permet la transmission de tout trafic IP, pas seulement des messages de protocole d'acheminement, entre les receveurs et les alimentations. Les receveurs peuvent acheminer des datagrammes sur l'Internet en utilisant l'alimentation ou receveur le plus convenable comme prochain bond. Les administrateurs peuvent vouloir régler la métrique utilisée par leurs protocoles d'acheminement pour refléter dans les tableaux d'acheminement les caractéristiques asymétriques de la liaison, et ainsi diriger le trafic sur les chemins appropriés.

Les alimentations et les receveurs peuvent mettre en œuvre un acheminement de diffusion groupée et donc des acheminements dynamiques de diffusion groupée peuvent être effectués sur la liaison unidirectionnelle. Cependant les questions en rapport avec l'acheminement de diffusion groupée (par exemple, la configuration du protocole) ne sont pas traitées dans le présent document.

9.3 Adaptabilité

Le protocole DTCP ne génère pas beaucoup de trafic quel que soit le nombre de nœuds. Le problème avec un grand nombre de nœuds n'est pas avec ce protocole mais avec des questions plus générales comme le nombre maximum de nœuds qui peuvent être connectés à une liaison. Ceci sort du domaine d'application de ce document.

10. Considérations en rapport avec l'IANA

L'IANA a réservé l'adresse 224.0.0.36 comme adresse de diffusion groupée des "annonces DTCP" comme défini à la Section 7.

L'IANA a réservé l'accès udp 652 pour le HELLO_PORT comme défini à la Section 7.

11. Considérations pour la sécurité

De nombreuses technologies de liaison unidirectionnelle sont caractérisées par la facilité avec laquelle le contenu de la liaison peut être reçu. Si des informations sensibles ou précieuses sont envoyées, des mécanismes de sécurité de couche liaison sont une mesure appropriée. Pour le protocole UDLR lui-même, les adresses de point d'extrémité de tunnel d'alimentation, envoyées dans les messages HELLO, peuvent être considérées comme sensibles. Dans de tels cas, des mécanismes de sécurité de couche liaison peuvent être utilisés.

La sécurité dans un réseau qui utilise le mécanisme de tunnelage de couche liaison devrait être relativement similaire à la sécurité dans un réseau IPv4 normal. Cependant, comme le mécanisme de tunnelage de couche liaison exige l'utilisation de tunnels, cela introduit un potentiel d'accès non autorisé au service. En particulier, l'utilisation d'un leurre ARP et IP est une menace potentielle parce que les nœuds peuvent n'être pas autorisés à tunneler les paquets. Ceci peut être contré par l'authentification de tous les tunnels. Le mécanisme d'authentification n'est pas spécifié dans le présent document ; il peut prendre place soit dans le protocole IP de livraison (par exemple., AH [RFC2402]) soit dans un protocole d'authentification intégré au mécanisme de tunnelage.

À un niveau supérieur, un receveur peut n'être pas autorisé à fournir des informations d'acheminement même si il est connecté à la liaison unidirectionnelles Afin d'empêcher que des receveurs non autorisés fournissent de fausses informations d'acheminement, les protocoles d'acheminement qui fonctionnent par dessus le mécanisme de tunnelage de couche liaison DOIVENT utiliser des mécanismes d'authentification lorsque ils sont disponibles.

12. Remerciements

Nous tenons à remercier Tim Gleeson (Cisco Japon) de ses précieux travaux d'édition et de ses apports techniques durant la phase de finalisation du document.

Nous souhaitons aussi remercier Patrick Cipiere (UDcast) de ses apports précieux concernant la conception du mécanisme d'encapsulation.

Nous tenons aussi à remercier de leur participation Akihiro Tosaka (IMD), Akira Kato (Tokyo Univ.), Hitoshi Asaeda (IBM/ITS), Hiromi Komatsu (JSAT), Hiroyuki Kusumoto (Keio Univ.), Kazuhiro Hara (Sony), Kenji Fujisawa (Sony), Mikiyo Nishida (Keio Univ.), Noritoshi Demizu (Sony CSL), Jun Murai (Keio Univ.), Jun Takei (JSAT) et Harri Hakulinen (Nokia).

Appendice A Conformité et interopérabilité

Le présent document décrit un mécanisme pour émuler la connexité bidirectionnelle entre des nœuds qui sont directement connectés par une liaison unidirectionnelles L'applicabilité sur divers équipements et environnements est assurée en permettant un choix entre plusieurs paramètres de système de clés.

Donc, pour assurer l'interopérabilité des équipements, il ne suffit pas de prétendre être simplement conforme au mécanisme défini ici. Un profil d'usage pour un environnement particulier va exiger la définition de plusieurs paramètres :
- le format de MAC utilisé,
- le mécanisme de tunnelage à utiliser (GRE est recommandé),
- l'indication de "type de tunnel" si GRE n'est pas utilisé.

Par exemple, un système peut prétendre mettre en œuvre "le mécanisme de tunnelage de couche liaison pour liaisons unidirectionnelles, en utilisant IEEE 802 LLC, et l'encapsulation GRE pour les tunnels."

Appendice B Plan de transition d'adresse d'annonce DTCP

Certains des plus anciens receveurs écoutent les annonces DTCP sur l'adresse de diffusion groupée 224.0.1.124 (la "vieille adresse d'annonces DTCP"). Afin de prendre en charge ces receveurs traditionnels, les alimentations DEVRAIENT être configurables de façon à envoyer toutes les annonces simultanément aux deux adresses "annonces DTCP", et "vieille annonce DTCP". Le réglage par défaut est d'envoyer les annonces juste à l'adresse "annonces DTCP".

Afin d'encourager au plan de transition, les "vieilles" alimentations DOIVENT être mises à jour pour envoyer les annonces DTCP comme défini dans cette section. Le nombre de "vieilles" alimentations déployées à l'origine est relativement faible et donc, la mise à jour devrait être assez facile. Les "nouveaux" receveurs prennent seulement en charge les "nouvelles" alimentations, c'est-à-dire qu'elles écoutent les annonces DTCP sur l'adresse "annonces DTCP".

Références

[RFC0826] D. Plummer, "Protocole de résolution d'adresses Ethernet : conversion des adresses de protocole réseau en adresses Ethernet à 48 bits pour transmission sur un matériel Ethernet", STD 37, novembre 1982.

[RFC1112] S. Deering, "Extensions d'hôte pour diffusion groupée sur IP", STD 5, août 1989. *(Màj par la* RFC2236*)*

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

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

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

[RFC2453] G. Malkin, "RIP version 2", STD 56, novembre 1998. *(Mise à jour par la* RFC4822*)*

[RFC2780] S. Bradner et V. Paxson, "Lignes directrices pour les allocations par l'IANA des valeurs du protocole Internet

et des en-têtes qui s'y rapportent", BCP 37, mars 2000.

[RFC2784] D. Farinacci, T. Li, S. Hanks, D. Meyer et P. Traina, "Encapsulation d'acheminement générique (GRE)", mars 2000.

Adresse des auteurs

Emmanuel Duros
UDcast
1681, route des Dolines
Les Taissounieres - BP 355
06906 Sophia-Antipolis Cedex
France
téléphone : +33 4 93 00 16 60
Fax : +33 4 93 00 16 61
mél : Emmanuel.Duros@UDcast.com

Walid Dabbous
INRIA Sophia Antipolis
2004, Route des Lucioles BP 93
06902 Sophia Antipolis
France
téléphone : +33 4 92 38 77 18
Fax : +33 4 92 38 79 78
mél : Walid.Dabbous@inria.fr

Hidetaka Izumiyama
JSAT Corporation
Toranomon 17 Mori Bldg.5F
1-26-5 Toranomon, Minato-ku
Tokyo 105
Japan
téléphone : +81-3-5511-7568
Fax : +81-3-5512-7181
mél : izu@jsat.net

Noboru Fujii
Sony Corporation
2-10-14 Osaki, Shinagawa-ku
Tokyo 141
Japan
téléphone : +81-3-3495-3092
Fax : +81-3-3495-3527
mél : fujii@dct.sony.co.jp

Yongguang Zhang
HRL
RL-96, 3011 Malibu Canyon Road
Malibu, CA 90265,
USA
téléphone : 310-317-5147
Fax : 310-317-5695
mél : ygz@hrl.com

Déclaration complète de droits de reproduction

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