Groupe de travail Réseau
Request for Comments : 3931
Catégorie : En cours de normalisation
Traduction Claude Brière de L'Isle

J. Lau, éditeur
M. Townsley, Cisco Systems
I. Goyret, Lucent Technologies
mars 2005

Protocole de tunnelage de couche deux - version 3 (L2TPv3)

Statut du présent mémoire
Le présent document spécifie un protocole en cours de normalisation de l'Internet pour la communauté de l'Internet, et appelle à des discussions et suggestions pour son amélioration. Prière de se référer à l'édition en cours des "Normes officielles des protocoles de l'Internet" (STD 1) pour connaître l'état de la normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

Résumé
Le présent document décrit la "version 3" du protocole de tunnelage de couche deux (L2TPv3, *Layer Two Tunneling Protocol*). L2TPv3 définit le protocole de contrôle de base et d'encapsulation pour tunneler plusieurs connexions de couche 2 entre deux nœuds IP. Des documents supplémentaires détaillent les spécificités de chaque type de liaison de données émulé.

Table des Matières

1. Introduction

Le protocole de tunnelage de couche deux (L2TP) donne un mécanisme dynamique pour tunneler les "circuits" de couche 2 (L2) à travers un réseau de données en mode paquet (par exemple, sur IP). L2TP, comme défini à l'origine dans la RFC2661, est une méthode standard pour tunneler des sessions du protocole point à point (PPP) [RFC1661]. L2TP a depuis été adopté pour tunneler un certain nombre d'autres protocoles de couche 2. Afin de fournir une plus grande modularité, le présent document décrit le protocole L2TP de base, indépendamment de la charge utile L2 qui est tunnelée.

Le protocole L2TP de base défini dans le présent document consiste en (1) le protocole de contrôle pour la création dynamique, la maintenance, et la suppression des sessions L2TP, et (2) l'encapsulation des données L2TP en flux de données de couche 2 multiplexés et démultiplexés entre deux nœuds L2TP à travers un réseau IP. La publication de documents supplémentaires est attendue pour chaque type d'émulation de liaison de données L2 (autrement dit de type pseudo filaire) pris en charge par L2TP (c'est-à-dire, PPP, Ethernet, relais de trame, etc.). Ces documents contiendront tous les détails de type pseudo filaire spécifiques qui sortent du domaine d'application de la présente spécification de base.

Lorsque la désignation entre L2TPv2 et L2TPv3 est nécessaire, L2TP comme défini dans la RFC2661 sera appelé "L2TPv2", correspondant à la valeur du champ Version d'un en-tête L2TP. (La transmission de couche 2 (L2F, *Layer 2 Forwarding* [RFC2341] a été définie comme "version 1".) Parfois, L2TP comme défini dans le présent document sera appelé "L2TPv3". Autrement, l'acronyme "L2TP" se réfère à L2TPv3 ou L2TP en général.

1.1 Changements par rapport à la RFC 2661

Beaucoup des constructions du protocole décrites dans le présent document sont reprises de la RFC2661. Les changements incluent des précisions fondées sur des années d'interopérabilité et de développement de l'expérience ainsi que des

modifications pour améliorer le fonctionnement du protocole ou fournir une plus claire séparation par rapport à PPP. L'intention de ces modifications est de réaliser un meilleur équilibre entre la réutilisation du code, l'expérience de l'interopérabilité, et une évolution dirigée de L2TP lorsque il est appliqué à de nouvelles tâches.

Les différences notables entre L2TPv2 et L2TPv3 incluent ce qui suit :

Séparation de toutes les AVP, références, etc. se rapportant à PPP, incluant une portion de l'en-tête de données L2TP qui étaient spécifiques des besoins de PPP. Les constructions spécifiques de PPP sont décrites dans un autre document.

Passage d'un identifiant de session et d'un identifiant de tunnel de 16 bits à respectivement un identifiant de session et un identifiant de connexion de contrôle de 32 bits.

Extension du mécanisme d'authentification de tunnel pour couvrir le message de contrôle entier plutôt que juste une portion de certains messages.

Les détails de ces changements et une recommandation pour la transition vers L2TPv3 sont discutés au paragraphe 4.7.

1.2 Spécification des exigences

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 la [RFC2119].

1.3. Terminologie

Paire attribut-valeur (AVP, *Attribute Value Pair*)
C'est l'enchaînement de longueur variable d'un attribut unique (représenté par un entier) un champ Longueur, et une valeur contenant la valeur réelle identifiée par l'attribut. Zéro, une ou plusieurs AVP constituent le corps des messages de contrôle, qui sont utilisés pour l'établissement, la maintenance, et la suppression des connexions de contrôle. Cette construction de base est parfois appelée un triplet type-longueur-valeur (TLV, *Type-Length-Value*) dans certaines spécifications. (Voir aussi connexion de contrôle, message de contrôle.)

Appel *(Call)* (circuit ouvert) *(Circuit Up)*
C'est l'action de faire passer un circuit sur un concentrateur d'accès L2TP (LAC, *L2TP Access Concentrator*) à un état "ouvert" ou "actif". Un appel peut être établi de façon dynamique à travers des propriétés de signalisation (par exemple, un appel entrant ou sortant à travers le réseau téléphonique public commuté (RTPC)) ou configuré de façon statique (par exemple, en provisionnant un circuit virtuel sur une interface). Un appel se définit par ses propriétés (par exemple, type d'appel, numéro demandé, etc.) et son trafic de données. (Voir aussi Circuit, Session, Appel entrant, Appel sortant, Demande d'appel sortant.)

Circuit
Terme général qui identifie une connexion de couche 2 parmi une large gamme. Un circuit peut être de nature virtuelle (par exemple, un PVC ATM, un VLAN IEEE 802, ou une session L2TP) ou il peut avoir une corrélation directe avec une couche physique (par exemple, une ligne de série RS-232). Les circuits peuvent être configurés de façon statique avec une durée d'activité relativement longue, ou établis de façon dynamique avec de la signalisation pour gouverner l'établissement, la maintenance, et la suppression du circuit. Pour les besoins du présent document, un circuit configuré de façon statique est considéré comme étant essentiellement le même qu'un très simple circuit dynamique de longue durée. (Voir aussi: Appel, Système distant.)

Client (Voir Système distant.)

Connexion de contrôle
Une connexion de contrôle L2TP est un canal de contrôle fiable qui est utilisé pour établir, maintenir, et libérer des sessions L2TP individuelles aussi bien que la connexion de contrôle elle-même. (Voir aussi Message de contrôle, Canal de données.)

Message de contrôle
C'est un message L2TP utilisé par la connexion de contrôle. (Voir aussi Connexion de contrôle.)

Message de données
Message utilisé par le canal de données. (Autrement dit, un paquet de données, Voir aussi Canal de données.)

Canal de données
C'est le canal pour le trafic de données encapsulé dans L2TP qui passe entre deux LCCE sur un réseau de commutation de paquets (c'est-à-dire, IP). (Voir aussi Connexion de contrôle, Message de données.)

Appel entrant
Action de recevoir un appel (événement de circuit ouvert) sur un LAC. L'appel peut avoir été passé par un système distant (par exemple, un appel téléphonique sur le RTPC) ou il peut avoir été déclanché par un événement local (par exemple, le trafic intéressant acheminé à une interface virtuelle). Un appel entrant qui a besoin d'être tunnelé (comme déterminé par le LAC) résulte en la génération d'un message ICRQ L2TP. (Voir aussi Appel, Appel sortant, Demande d'appel sortant.)

Concentrateur d'accès L2TP (LAC, *L2TP Access Concentrator*)
Si un point d'extrémité de connexion de contrôle L2TP (LCCE, *L2TP Control Connection Endpoint*) est utilisé pour interconnecter une session L2TP directement à une liaison de données, on s'y réfère comme à un concentrateur d'accès L2TP (LAC). Un LCCE peut agir à la fois comme serveur réseau L2TP (LNS, *L2TP Network Server*) pour certaines sessions et comme LAC pour d'autres, de sorte que ces termes ne doivent être utilisés que dans le contexte d'un certain ensemble de sessions sauf si le LCCE est en fait à un seul objet pour une certaine topologie. (Voir aussi LCCE, LNS.)

Point d'extrémité de connexion de contrôle L2TP (LCCE)
C'est un nœud L2TP qui existe à l'une et l'autre extrémité d'une connexion de contrôle L2TP. Il PEUT aussi être appelé un LAC ou un LNS, selon que les trames tunnelées sont traitées dans la liaison de données (LAC) ou dans la couche réseau (LNS). (Voir aussi LAC, LNS.)

Serveur réseau L2TP (LNS, *L2TP Network Server*)
Si une certaine session L2TP se termine au nœud L2TP et si le paquet encapsulé de couche réseau (L3) est traité sur une interface virtuelle, on appelle ce nœud L2TP un serveur réseau L2TP (LNS, *L2TP Network Server*). Un certain LCCE peut agir à la fois comme LNS pour certaines sessions et comme LAC pour d'autres, de sorte que ces termes ne doivent être utilisés que dans le contexte d'un certain ensemble de sessions sauf si le LCCE est en fait à un seul objet pour une certaine topologie. (Voir aussi LCCE, LAC.)

Appel sortant
C'est l'action de passer un appel par un LAC, normalement en réponse à une politique dirigée par l'homologue dans une demande d'appel sortant. (Voir aussi Appel, Appel entrant, Demande d'appel sortant.)

Demande d'appel sortant
Demande envoyée à un LAC pour passer un appel sortant. La demande contient des informations spécifiques non connues à priori par le LAC (par exemple, un numéro à composer). (Voir aussi Appel, Appel entrant, Appel sortant.)

Réseau à commutation de paquets (PSN, *Packet-Switched Network*)
Réseau qui utilise la technologie de la commutation de paquets pour la livraison des données. Pour L2TPv3, cette couche est principalement IP. D'autres exemples incluent MPLS, le relais de trame, et ATM.

Homologue
Lorsque utilisé dans un contexte de L2TP, homologue se réfère à l'extrémité distante d'une connexion de contrôle L2TP (c'est-à-dire, le LCCE distant). Un homologue de LAC peut être soit un LNS soit un autre LAC. De même, l'homologue d'un LNS peut être soit un LAC, soit un autre LNS. (Voir aussi LAC, LCCE, LNS.)

Pseudo filaire (PW, *Pseudowire*)
Circuit émulé lorsque il traverse un PSN. Il y a un pseudo filaire par session L2TP. (Voir aussi Réseau à commutation de paquets, Session.)

Type pseudo filaire
Type de charge utile qui est portée au sein d'une session L2TP. Les exemples incluent PPP, Ethernet, et le relais de trame. (Voir aussi Session.)

Système distant
Système ou routeur distant connecté par un circuit à un LAC.

Session
Une session L2TP est l'entité qui est créée entre deux LCCE afin d'échanger les paramètres pour établir et maintenir une connexion L2 émulée. Plusieurs sessions peuvent être associées à une seule connexion de contrôle.

Message au corps de longueur zéro (ZLB, *Zero-Length Body*)
Message de contrôle qui a seulement un en-tête L2TP. Les messages ZLB ne sont utilisés que pour accuser réception des

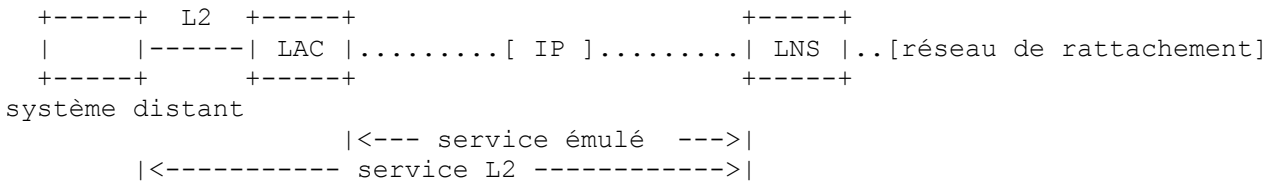
messages sur la connexion de contrôle L2TP fiable. (Voir aussi Message de contrôle.)

2. Topologie

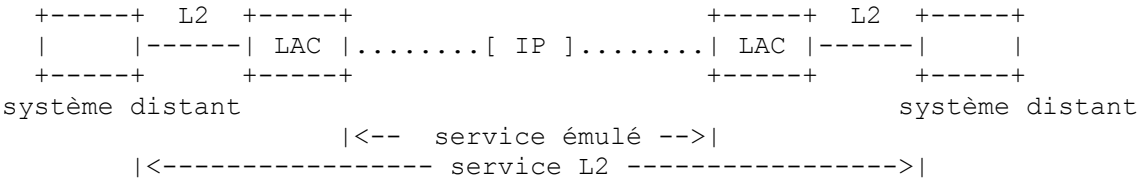
L2TP opère entre deux points d'extrémité de connexion de contrôle L2TP (LCCE), tunnelant le trafic à travers un réseau de paquets. Il y a trois modèles prédominants de tunnelage dans lesquels fonctionne L2TP : LAC-LNS (ou vice versa), LAC-LAC, et LNS-LNS. Ces modèles sont illustrés ci-dessous par des diagrammes. (Les lignes en pointillés désignent les connexions réseau. Les lignes continues désignent les circuits de connexion.)

Figure 2.0 : Modèles de référence L2TP

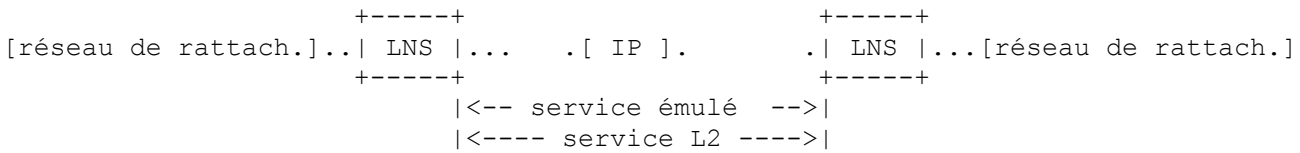
(a) Modèle de référence LAC-LNS : sur un côté, le LAC reçoit le trafic d'un circuit de couche 2, qu'il transmet via L2TP à travers IP ou un autre réseau fondé sur le paquet. De l'autre côté, un LNS termine logiquement le circuit de couche 2 en local et achemine le trafic réseau au réseau de rattachement. L'action d'établissement de session est pilotée par le LAC (comme un appel entrant) ou par le LNS (comme un appel sortant).



(b) Modèle de référence LAC-LAC : dans ce modèle, les deux LCCE sont des LAC. Chaque LAC transmet le trafic circuit du système distant au LAC homologue en utilisant L2TP, et vice versa. Dans sa plus simple forme, un LAC agit comme une simple interconnexion entre un circuit à un système distant et une session L2TP. Ce modèle implique normalement un établissement symétrique ; c'est-à-dire que l'un et l'autre côté de la connexion peut initier une session à tout moment (ou simultanément, auquel cas un mécanisme de départage est utilisé).



(c) Modèle de référence LNS-LNS : ce modèle a deux LNS comme LCCE. Un événement de niveau utilisateur, généré par le trafic, ou signalé, pilote normalement l'établissement de session à partir d'un côté du tunnel. Par exemple, un tunnel généré à partir d'un PC par un usager, ou automatiquement par un équipement dans les locaux de l'abonné.



Note : dans L2TPv2, le tunnelage piloté par l'usager de ce type est souvent appelé un "tunnelage volontaire" [RFC2809]. De plus, un LNS qui agit au titre d'un paquetage logiciel sur un hôte est parfois appelé un "LAC client" [RFC2661].

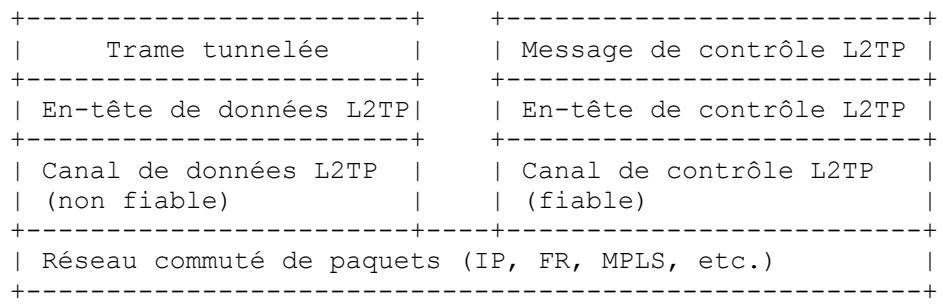
3. Vue d'ensemble du protocole

L2TP se compose de deux types de messages, les messages de contrôle et les messages de données (parfois appelés respectivement des "paquets de contrôles" et des "paquets de données"). Les messages de contrôle sont utilisés à l'établissement, la maintenance, et la suppression des connexions de contrôle et des sessions. Ces messages utilisent un canal de contrôle fiable au sein de L2TP pour garantir la livraison (voir les détails au paragraphe 4.2). Les messages de données sont utilisés pour encapsuler le trafic L2 qui est porté sur la session L2TP. À la différence des messages de contrôle, les messages de données ne sont pas retransmis lorsque survient une perte de paquet.

Le format des messages de contrôle L2TPv3 défini dans le présent document emprunte largement à L2TPv2. Ces messages de contrôle sont utilisés en conjonction avec les automates à états de protocole associés qui pilotent l'établissement dynamique, la

maintenance, et la suppression des sessions L2TP. Le format de message de données pour le tunnelage des paquets de données peut être utilisé avec ou sans le canal de contrôle L2TP, via configuration manuelle ou via d'autres méthodes de signalisation pour des informations de session L2TP préconfigurées ou réparties. L'utilisation du format de message de données L2TP avec d'autres méthodes de signalisation sort du domaine d'application du présent document.

Figure 3.0 : Structure L2TPv3



La Figure 3.0 décrit les relations entre messages de contrôle et messages de données sur, respectivement, les canaux de contrôle et de données L2TP. Les messages de données sont passés sur un canal de données non fiable, encapsulés par un en-tête L2TP, et envoyés sur un réseau commuté de paquets (PSN) comme IP, UDP, relais de trame, ATM, MPLS, etc. Les messages de contrôle sont envoyés sur un canal de contrôle L2TP fiable, qui fonctionne sur le même PSN.

L'établissement nécessaire pour tunneler une session avec L2TP consiste en deux étapes : (1) établir la connexion de contrôle, et (2) établir une session comme déclanchée par un appel entrant ou un appel sortant. Une session L2TP DOIT être établie avant que L2TP puisse commencer à transmettre des trames de session. Plusieurs sessions peuvent être liées à une seule connexion de contrôle, et plusieurs connexions de contrôle peuvent exister entre les deux mêmes LCCE.

3.1 Types de message de contrôle

L'AVP Type de message (voir au paragraphe 5.4.1) définit le type spécifique de message de contrôle envoyé.

Le présent document définit les types de message de contrôle suivants (voir aux paragraphes 6.1 à 6.15 les détails sur la construction et l'utilisation de chaque message) :

Gestion de la connexion de contrôle
0 (réservé)
1 (SCCRQ) *(Start-Control-Connection-Request)* demande de commencer la connexion de contrôle
2 (SCCRP) *(Start-Control-Connection-Reply)* réponse à commencer la connexion de contrôle
3 (SCCCN) *(Start-Control-Connection-Connected)* connexion de contrôle connectée
4 (StopCCN) *(Stop-Control-Connection-Notification)* notification d'arrêt de la connexion de contrôle
5 (réservé)
6 (HELLO) Hello
20 (ACK) Accusé de réception explicite

Gestion d'appel
7 (OCRQ) *(Outgoing-Call-Request)* demande d'appel sortant
8 (OCRP) *(Outgoing-Call-Reply)* réponse d'appel sortant
9 (OCCN) *(Outgoing-Call-Connected)* appel sortant connecté
10 (ICRQ) *(Incoming-Call-Request)* demande d'appel entrant
11 (ICRP) *(Incoming-Call-Reply)* réponse d'appel entrant
12 (ICCN) *(Incoming-Call-Connected)* appel entrant connecté
13 (réservé)
14 (CDN) *(Call-Disconnect-Notify)* notification d'appel déconnecté

Rapport d'erreur
15 (WEN) *(WAN-Error-Notify)* notification d'erreur de WAN

Rapport de changement d'état de la liaison
16 (SLI) *(Set-Link-Info)* informations sur l'établissement de la liaison

3.2 Formats d'en-tête L2TP

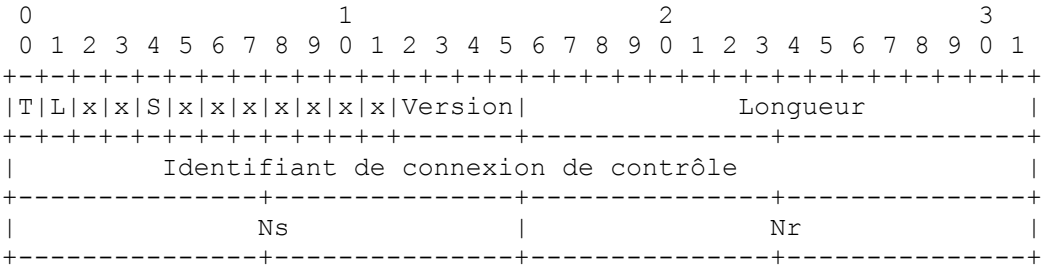
Ce paragraphe définit les formats d'en-tête pour les messages L2TP de contrôle et les messages L2TP de données. Toutes les valeurs sont placées dans leurs champs respectifs et envoyées dans l'ordre du réseau (octets de poids fort en premier).

3.2.1 En-tête de message de contrôle L2TP

L'en-tête de message de contrôle L2TP fournit des informations pour le transport fiable des messages qui gouvernent l'établissement, la maintenance, et la suppression des sessions L2TP. Par défaut, les messages de contrôle sont envoyés sur le support sous-jacent dans la bande avec les messages L2TP de données.

L'en-tête de message de contrôle L2TP est formaté comme suit :

Figure 3.2.1 : En-tête de message de contrôle L2TP



Le bit T DOIT être à 1, qui indique que c'est un message de contrôle.

Les bits L et S DOIVENT être à 1, qui indique que les champs Longueur et Numéro de séquences sont présents.

Les bits x sont réservés pour de futures extensions. Tous les bits réservés DOIVENT être à 0 sur les messages sortants et ignorés sur les messages entrants.

Le champ Version indique la version de l'en-tête de message de contrôle L2TP décrit dans le présent document. À l'envoi, ce champ DOIT être réglé à 3 pour tous les messages (sauf à fonctionner dans un environnement qui inclut aussi L2TPv2 [RFC2661] et/ou L2F [RFC2341], voir les détails au paragraphe 4.1).

Le champ Longueur indique la longueur totale du message en octets, toujours calculée depuis le début de l'en-tête du message de contrôle lui-même (commençant au bit T).

Le champ Identifiant de connexion de contrôle contient l'identifiant de la connexion de contrôle. Les connexions de contrôle L2TP sont nommées par des identifiants qui n'ont de signification que locale. C'est-à-dire que la même connexion de contrôle recevra des identifiants uniques de connexion de contrôle par chaque LCCE depuis l'espace de numéros d'identifiant de connexion de contrôle de chaque point d'extrémité. À ce titre, l'identifiant de connexion de contrôle dans chaque message est celui du receveur prévu, et non de l'envoyeur. Les identifiants de connexion de contrôle différents de zéro sont choisis et échangés comme des AVP d'identifiant de connexion de contrôle allouées durant la création d'une connexion de contrôle.

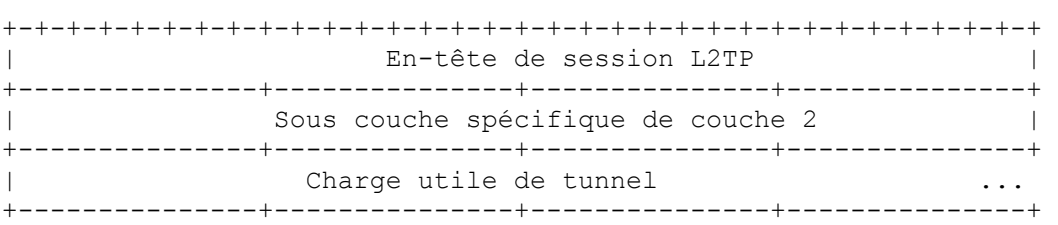
Ns indique le numéro de séquence pour ce message de contrôle, commençant à zéro et incrémenté de un (modulo 2**16) pour chaque message envoyé. Voir au paragraphe 4.2 plus d'informations sur l'utilisation de ce champ.

Nr indique le numéro de séquence attendu dans le prochain message de contrôle à recevoir. Donc, Nr est réglé au Ns du dernier message dans l'ordre reçu plus un (modulo 2**16). Voir au paragraphe 4.2 plus d'informations sur l'utilisation de ce champ.

3.2.2 Message de données L2TP

En général, un message L2TP de données consiste en (1) un en-tête de session, (2) une sous couche facultative spécifique de couche 2, et (3) la charge utile de tunnel, comme décrit ci-dessous.

Figure 3.2.2 : En-tête de message de données L2TP



L'en-tête de session L2TP est spécifique du PSN encapsulant sur lequel le trafic L2TP est livré. L'en-tête de session DOIT fournir (1) une méthode pour distinguer le trafic entre plusieurs sessions de données L2TP et (2) une méthode pour distinguer les messages de données des messages de contrôle.

Chaque type de PSN encapsulant DOIT définir son propre en-tête de session, identifiant clairement le format de l'en-tête et les paramètres nécessaires pour établir la session. Le paragraphe 4.1 définit deux en-têtes de session, un pour le transport sur UDP et un pour le transport sur IP.

La sous couche spécifique de couche 2 est une couche intermédiaire entre l'en-tête de session L2TP et le début de la trame tunnelée. Elle contient des champs de contrôle qui sont utilisés pour faciliter le tunnelage de chaque trame (par exemple, numéros de séquence ou fanions). La sous couche spécifique de couche 2 par défaut pour L2TPv3 est définie au paragraphe 4.6.

L'en-tête de message de données est suivi par la charge utile de tunnel, incluant tout tramage de couche 2 nécessaire, comme défini dans les documents d'accompagnement spécifiques des charges utiles.

3.3 Gestion de la connexion de contrôle

La connexion de contrôle L2TP traite l'établissement dynamique, la suppression et la maintenance des sessions L2TP et de la connexion de contrôle elle-même. La livraison fiable des messages de contrôle est décrite au paragraphe 4.2.

Ce paragraphe décrit les échanges typiques d'établissement et de suppression de connexion de contrôle. Il est important de noter que, dans les diagrammes qui suivent, le mécanisme de livraison fiable de message de contrôle existe indépendamment de l'automate à états L2TP. Par exemple, les messages d'accusé de réception explicite (ACK) peuvent être envoyés après tout message de contrôle indiqué dans les échanges ci-dessous si un accusé de réception n'est pas porté sur un message de contrôle ultérieur.

Les LCCE sont identifiés durant l'établissement de la connexion de contrôle par l'AVP Nom d'hôte, par l'AVP Identifiant de routeur, ou par une combinaison des deux (voir au paragraphe 5.4.3). L'identité d'un LCCE homologue est centrale pour choisir les paramètres de configuration appropriés (c'est-à-dire, l'intervalle de Hello, la taille de fenêtre, etc.) pour une connexion de contrôle, ainsi que pour déterminer comment établir les sessions associées au sein de la connexion de contrôle, la recherche de mot de passe pour l'authentification de la connexion de contrôle, le départage de niveau de connexion de contrôle, etc.

3.3.1 Établissement de la connexion de contrôle

L'établissement de la connexion de contrôle implique un échange des AVP qui identifient l'homologue et ses capacités.

Un échange de trois messages est utilisé pour établir la connexion de contrôle. Il se présente normalement comme suit:

LCCE A **LCCE B**
SCCRQ -->
<-- SCCRP
SCCCN -->

3.3.2 Suppression de la connexion de contrôle

La suppression de la connexion de contrôle peut être initiée par l'un ou l'autre LCCE et est accomplie par l'envoi d'un seul message de contrôle StopCCN. Au titre du mécanisme de livraison fiable du message de contrôle, le receveur d'un StopCCN DOIT envoyer un message ACK pour accuser réception du message et maintenir assez d'état de connexion de contrôle pour accepter de façon appropriée les retransmissions de StopCCN sur au moins un cycle complet de retransmission (en cas de perte du message ACK). La durée recommandée pour un cycle de retransmission complet est d'au moins 31 secondes (voir au paragraphe 4.2). Voici un exemple d'un échange typique de messages de contrôle :

LCCE A **LCCE B**
StopCCN -->
(Nettoyer)
(Attendre)
(Nettoyer)

Une mise en œuvre peut fermer une connexion de contrôle entière et toutes les sessions associées à la connexion de contrôle en envoyant le StopCCN. Donc, il n'est pas nécessaire de fermer chaque session individuellement lors de la fermeture de la connexion de contrôle entière.

3.4 Gestion de session

Après la réussite de l'établissement d'une connexion de contrôle, des sessions individuelles peuvent être créées. Chaque session correspond à un seul flux de données entre les deux LCCE. Cette section décrit les échanges typiques d'établissement et de suppression d'appel.

3.4.1 Établissement de session pour un appel entrant

Un échange de trois messages est utilisé pour établir la session. Voici une séquence d'événements typique :

LCCE A **LCCE B**
(appel détecté)
ICRQ -->
<-- ICRP
(appel accepté)
ICCN -->

3.4.2 Établissement de session pour un appel sortant

Un échange de trois messages est utilisé pour établir la session. Voici une séquence d'événements typique :

LCCE A **LCCE B**
<-- OCRQ
OCRP -->
(Effectuer l'opération d'appel)
OCCN -->
(Opération d'appel bien achevée)

3.4.3 Suppression de session

Une suppression de session peut être initiée par le LAC ou le LNS et est accomplie par l'envoi d'un message de contrôle CDN. Après la suppression de la dernière session, la connexion de contrôle PEUT être aussi supprimée (et l'est normalement). Voici un exemple d'échange typique de messages de contrôle :

LCCE A **LCCE B**
CDN -->
(Nettoyer)
(Nettoyer)

4. Fonctionnement du protocole

4.1 L2TP sur réseaux spécifiques de commutation de paquets

L2TP peut fonctionner sur diverses sortes de PSN. Deux modes sont décrits pour le fonctionnement sur IP, L2TP directement sur IP (voir au paragraphe 4.1.1) et L2TP sur UDP (voir au paragraphe 4.1.2). Les mises en œuvre de L2TPv3 DOIVENT prendre en charge L2TP sur IP et DEVRAIENT prendre en charge L2TP sur UDP pour une meilleure traversée de NAT et de pare-feu, et pour une migration plus facile à partir de L2TPv2.

L2TP sur d'autres PSN peut être défini, mais ses spécificités sortent du domaine d'application du présent document. Des exemples de L2TPv2 sur d'autres PSN incluent la [RFC3070] et la [RFC3355].

Les définitions de champs suivantes sont pour toutes les encapsulations d'en-tête de session L2TP.

Identifiant de session
Champ de 32 bits qui contient un identifiant différent de zéro pour une session. Les sessions L2TP sont désignées par des identifiants qui n'ont qu'une signification locale. C'est-à-dire que la même session logique va recevoir des identifiants de session différents par chaque extrémité de la connexion de contrôle pour la vie de la session. Lorsque la connexion de contrôle L2TP est utilisée pour un établissement de session, les identifiants de session sont choisis et échangés comme des AVP d'identifiant de session locaux durant la création d'une session. L'identifiant de session seul fournit le contexte nécessaire pour tout le traitement de paquet ultérieur, incluant la présence, la taille, et la valeur du mouchard, du type de sous couche spécifique de couche 2, et le type de charge utile tunnelée.

Mouchard
Le champ facultatif Mouchard contient une valeur de longueur variable (d'un maximum de 64 bits) utilisée pour vérifier l'association d'un message de données reçu avec la session identifiée par l'identifiant de session. Le mouchard DOIT être réglé à la valeur aléatoire configurée ou signalée pour cette session. Le mouchard fournit un niveau supplémentaire de garantie qu'un message de données a été dirigé sur la session appropriée par l'identifiant de session. Un mouchard bien choisi peut empêcher de mal diriger par inadvertance des paquets égarés avec des identifiants de session récemment réutilisés, des identifiants de session soumis à la corruption du paquet, etc. Le mouchard peut aussi fournir une protection contre des attaques spécifiques d'insertion de paquets malveillants, comme décrit au paragraphe 8.2.

Lorsque la connexion de contrôle L2TP est utilisée pour l'établissement de session, des valeurs de mouchard aléatoires sont choisies et échangées comme AVP de mouchards allouées durant la création de session.

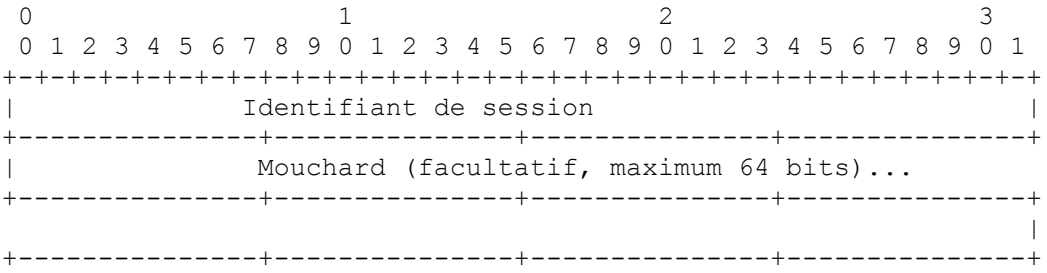
4.1.1 2TPv3 sur IP

L2TPv3 sur IP (les deux versions) utilise l'identifiant de protocole IP alloué par l'IANA de 115.

4.1.1.1 En-tête de session L2TPv3 sur IP

À la différence de L2TP sur UDP, l'en-tête de session L2TPv3 sur IP est libre de toute restriction imposée par la coexistence avec L2TPv2 et L2F. À ce titre, le format d'en-tête a été conçu pour optimiser le traitement de paquet. Le format d'en-tête de session suivant est utilisé lors du fonctionnement de L2TPv3 sur IP:

Figure 4.1.1.1 : En-tête de session L2TPv3 sur IP



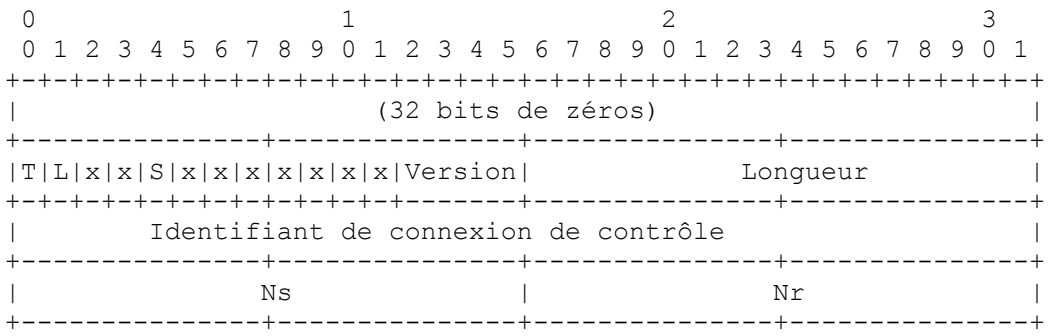
Les champs Identifiant de session et Mouchard sont définis au paragraphe 4.1. L'identifiant de session de zéro est réservé pour être utilisé par les messages L2TP de contrôle (voir au paragraphe 4.1.1.2).

4.1.1.2 Trafic L2TP de contrôle et de données sur IP

À la différence de L2TP sur UDP, qui utilise le bit T pour distinguer entre les paquets L2TP de contrôle et de données, L2TP sur IP utilise l'identifiant de session réservé de zéro (0) lors de l'envoi des messages de contrôle. On suppose que la vérification de l'identifiant de session de zéro est plus efficace – à la fois en taille d'en-tête pour les paquets de données et en vitesse de traitement pour distinguer entre messages de contrôle et de données – que de vérifier un seul bit.

L'en-tête entier de message de contrôle sur IP, incluant l'identifiant de session à zéro apparaît comme suit :

Figure 4.1.1.2 : En-tête L2TPv3 de message de contrôle sur IP



Les champs sont définis au paragraphe 3.2.1. Noter que le champ Longueur est toujours calculé à partir du début de l'en-tête du message de contrôle, commençant par le bit T. Il N'inclut PAS le "(32 bits de zéros)" montré ci-dessus.

En fonctionnement direct sur IP, les paquets L2TP perdent la capacité à tirer parti de la somme de contrôle UDP comme simple vérification d'intégrité du paquet, qui est un souci particulier des messages de contrôle L2TP. L'authentification du message de contrôle (voir au paragraphe 4.3) même avec un champ Mot de passe vide, fournit une vérification d'intégrité du paquet suffisante et DEVRAIT toujours être activée.

4.1.2 L2TP sur UDP

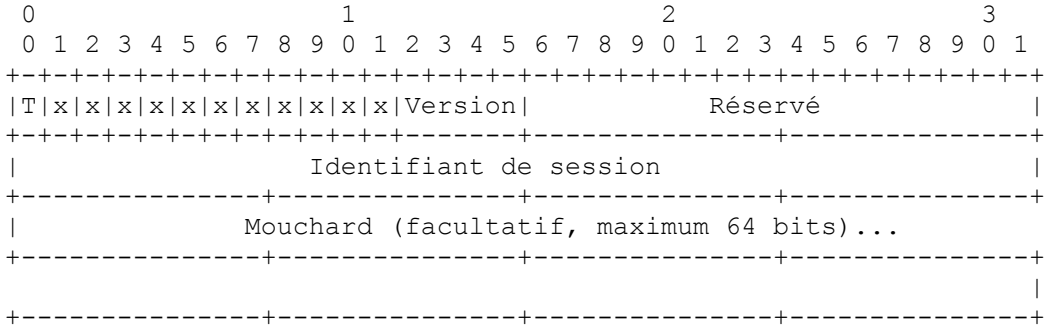
L2TPv3 sur UDP doit considérer les autres protocoles de tunnelage de couche 2 qui peuvent fonctionner dans le même environnement, incluant L2TPv2 [RFC2661] et L2F [RFC2341].

Bien qu'il y ait un gain d'efficacité à faire fonctionner L2TP directement sur IP, il peut aussi y avoir des effets collatéraux. Par exemple, L2TP sur IP n'est pas aussi favorable aux NAT que L2TP sur UDP.

4.1.2.1 En-tête de session L2TP sur UDP

Le format d'en-tête de session suivant est utilisé lors du fonctionnement de L2TPv3 sur UDP :

Figure 4.1.2.1 : En-tête de session L2TPv3 sur UDP



Le bit T DOIT être réglé à 0, qui indique que c'est un message de données.

Les bits x et le champ Réservé sont réservés pour de futures extensions. Toutes les valeurs réservées DOIVENT être réglées à 0 sur les messages sortants et ignorées sur les messages entrants.

Le champ Version DOIT être réglé à 3, indiquant un message L2TPv3.

Noter que les bits initiaux 1, 4, 6, et 7 ont une signification dans L2TPv2 [RFC2661], et sont déconseillés et marqués comme réservés dans L2TPv3. Donc, pour le mode UDP sur un système qui supporte les deux versions de L2TP, il est important que le champ Version soit inspecté avant de déterminer la version de l'en-tête et d'agir sur aucun de ces bits.

Les champs Identifiant de session et Mouchard sont comme défini au paragraphe 4.1.

4.1.2.2 Choix de l'accès UDP

La méthode pour le choix de l'accès UDP défini dans ce paragraphe est identique à celle définie pour L2TPv2 [RFC2661].

Lors de la négociation d'une connexion de contrôle sur UDP, les messages de contrôle DOIVENT être envoyés comme des datagrammes UDP en utilisant l'accès UDP enregistré de 1701 [RFC1700]. L'initiateur d'une connexion de contrôle L2TP prend un accès de source UDP disponible (qui peut être ou non 1701) et envoie à l'adresse de destination désirée à l'accès 1701. Le receveur prend un accès libre sur son propre système (qui peut être ou non 1701) et envoie sa réponse à l'accès UDP et adresse de l'initiateur, en réglant son propre accès de source à l'accès libre qu'il a trouvé.

Tout trafic suivant associé à cette connexion de contrôle (trafic de contrôle ou trafic de données provenant d'une session établie par cette connexion de contrôle) doit utiliser les mêmes accès UDP.

Il a été suggéré que si le receveur choisit un accès de source arbitraire (par opposition à l'utilisation de l'accès de destination mentionné dans le paquet qui initie la connexion de contrôle, c'est-à-dire, 1701) cela peut rendre plus difficile pour L2TP de traverser certains appareils de NAT. Les mises en œuvre devraient considérer les implications potentielles de cette capacité avant de choisir un accès de source arbitraire. Un appareil de NAT qui peut passer le trafic TFTP avec des accès UDP différents devrait être capable de passer le trafic UDP L2TP car les deux protocoles emploient des politiques similaires par rapport au choix de l'accès UDP.

4.1.2.3 Somme de contrôle UDP

Les trames tunnelées que porte L2TP ont souvent leur propres sommes de contrôle ou vérifications d'intégrité, rendant la somme de contrôle UDP redondante pour beaucoup des contenus des messages de données L2TP. Donc, les sommes de contrôle UDP PEUVENT être désactivées afin de réduire la charge de traitement de paquet associée aux points d'extrémité L2TP.

L'en-tête L2TP elle-même n'a pas sa propre somme de contrôle ou vérification d'intégrité. Cependant, l'utilisation de la paire Identifiant de session L2TP et Mouchard protège contre l'acceptation d'un message L2TP de données si une corruption de l'identifiant de session ou du mouchard associé s'est produite. Lorsque la sous couche spécifique de couche 2 est présente dans l'en-tête L2TP, il n'y a pas de vérification d'intégrité incorporée pour les informations qui y sont contenues si les sommes de contrôle UDP ou d'autres vérifications d'intégrité ne sont pas employées. IPsec (voir au paragraphe 4.1.3) peut être utilisé pour une forte protection de l'intégrité du contenu entier des messages L2TP de données.

Les sommes de contrôle UDP DOIVENT être activées pour les messages L2TP de contrôle.

4.1.3 L2TP et IPsec

Le canal de données L2TP n'assure aucune sécurité cryptographique. Si le canal de données L2TP fonctionne sur un réseau public ou un réseau IP qui n'est pas de confiance où on se soucie de la confidentialité des données L2TP ou si on s'attend à des attaques sophistiquées contre L2TP, IPsec [RFC2401] DOIT être disponible pour sécuriser le trafic L2TP.

L2TP sur UDP ou L2TP sur IP peuvent l'un et l'autre être sécurisés avec IPsec. La [RFC3193] définit la méthode recommandée pour sécuriser L2TPv2. L2TPv3 possède des caractéristiques identiques à l'égard d'IPsec à celles de L2TPv2 lorsque il fonctionne sur UDP et les mises en œuvre DOIVENT suivre la même recommandation. Lorsque il fonctionne directement sur IP, la [RFC3193] s'applique encore, bien que les références aux accès UDP de source et de destination (en particulier, celles de la Section 4, "Détails du filtrage IPsec pour la protection de L2TP") puissent être ignorées. Bien sûr, les sélecteurs utilisés pour identifier le trafic L2TPv3 sont simplement les adresses IP de source et de destination pour les points d'extrémité du tunnel avec le type de protocole IP L2TPv3 de 115.

En plus de la sécurité du transport IP, IPsec définit un mode de fonctionnement qui permet le tunnelage des paquets IP. Le chiffrement et l'authentification au niveau du paquet fournis par le mode tunnel IPsec et ceux fournis par L2TP sécurisé avec IPsec assurent un niveau équivalent de sécurité pour ces exigences.

IPsec définit aussi des caractéristiques de contrôle d'accès qui sont exigées des mises en œuvre conformes à IPsec. Ces caractéristiques permettent le filtrage des paquets fondé sur les caractéristiques des couches réseau et transport comme l'adresse IP, l'accès, etc. Dans le modèle de tunnelage L2TP, un filtrage analogue peut être effectué à la couche réseau au dessus de L2TP. Ces caractéristiques de contrôle d'accès de couche réseau peuvent être traitées à un LCCE via des caractéristiques d'autorisation spécifiques du fabricant, ou à la couche réseau elle-même en utilisant le mode transport IPsec de bout en bout entre les hôtes communicants. Les exigences pour les mécanismes de contrôle d'accès ne font pas partie de la spécification L2TP, et à ce titre sortent du domaine d'application du présent document.

La protection du flux de paquets L2TP par IPsec protège à son tour aussi les données au sein des paquets de la session tunnelée

pendant qu'ils sont transportés d'un LCCE à l'autre. Une telle protection ne doit pas être considérée comme substitution à la sécurité de bout en bout entre les hôtes ou applications communicants.

4.1.4 Questions de fragmentation IP

La fragmentation et le réassemblage dans les équipements de réseau exige généralement des ressources significativement supérieures à celles pour l'envoi ou la réception d'un paquet à l'unité. À ce titre, la fragmentation et le réassemblage devraient être évités chaque fois que possible. Les solutions idéales pour éviter la fragmentation incluent une configuration appropriée et la gestion des tailles de MTU entre le système distant, le LCCE, et le réseau IP, ainsi que des mesures d'adaptation qui fonctionnent avec l'hôte d'origine (par exemple, les [RFC1191], [RFC1981]) pour réduire les tailles de paquet à la source.

Un LCCE PEUT fragmenter un paquet avant l'encapsulation dans L2TP. Par exemple, si un paquet IPv4 arrive à un LCCE d'un système distant qui, après encapsulation avec son tramage associé, L2TP, et IP, ne tient pas dans la MTU de chemin disponible vers son LCCE homologue, le LCCE local peut effectuer une fragmentation IPv4 sur le paquet avant l'encapsulation dans le tunnel. Cela crée deux (ou plus) paquets L2TP, chacun portant un fragment IPv4 avec son tramage associé. Cela a en fin de compte pour effet de placer la charge de la fragmentation sur le LCCE, tandis que le réassemblage se fait chez l'hôte IPv4 de destination.

Si un paquet IPv6 arrive à un LCCE d'un système distant qui, après encapsulation avec le tramage associé, L2TP et IP, ne tient pas dans la MTU de chemin disponible vers son homologue L2TP, le paragraphe 7.1 de la spécification générique de tunnelage de paquet [RFC2473] DEVRAIT être suivi. Dans ce cas, le LCCE devrait soit envoyer un message ICMP Paquet trop gros à la source des données, soit fragmenter le paquet L2TP/IP résultant (pour réassemblage par l'homologue L2TP).

Si la quantité de trafic qui exige la fragmentation et le réassemblage est assez légère, ou si il y a des mécanismes suffisamment optimisés aux points d'extrémité du tunnel, la fragmentation du paquet L2TP/IP peut être suffisante pour s'accommoder des MTU discordantes qui ne peuvent pas être gérées par des moyens plus efficaces. Cette méthode émule effectivement une MTU supérieure entre les points d'extrémité du tunnel et devrait fonctionner pour tous les types de paquet encapsulé en couche 2. Noter que IPv6 ne prend pas en charge la fragmentation "au vol" des paquets de données. Donc, à la différence de IPv4, la MTU du chemin vers un homologue L2TP doit être connue à l'avance (ou la MTU minimum IPv6 de 1280 octets utilisée en dernier ressort) afin que la fragmentation IPv6 puisse survenir au LCCE.

En résumé, tenter de contrôler la MTU de source en communiquant avec l'hôte d'origine, en forçant une MTU suffisante sur le chemin entre les LCCE homologues pour tunneler une trame à partir de toute autre interface sans fragmentation, en fragmentant les paquets IP avant l'encapsulation avec L2TP/IP, ou en fragmentant le paquet L2TP/IP résultant entre les points d'extrémité de tunnel, sont toutes des méthodes valides pour gérer les discordances de MTU. Certaines sont clairement meilleures que d'autres selon le déploiement considéré. Par exemple, une application de surveillance passive qui utilise L2TP ne souhaiterait certainement pas avoir des messages ICMP envoyés à une source de trafic. De plus, si les liaisons qui connectent un ensemble de LCCE ont une très forte MTU (par exemple, SDH/SONET) et si on sait que toutes les liaisons qui sont tunnelées par L2TP ont une MTU plus petite (par exemple, 1500 octets) alors aucune fragmentation et réassemblage IP activé sur les LCCE participants ne sera jamais utilisée. Une mise en œuvre DOIT au moins utiliser une des méthodes décrites dans cette section pour gérer les MTU discordantes, sur la base d'une considération attentive de la façon dont le produit final sera déployé.

Les méthodes spécifiques de L2TP de fragmentation et réassemblage, qui peuvent dépendre ou non des caractéristiques du type de liaison tunnelée (par exemple, un paquetage judicieux de cellules ATM) peuvent être définies aussi, mais ces méthodes sortent du domaine d'application du présent document.

4.2 Livraison fiable des messages de contrôle

L2TP fournit un service fiable de livraison de niveau inférieur pour tous les messages de contrôle. Les champs Nr et Ns de l'en-tête du message de contrôle (voir au paragraphe 3.2.1) appartiennent à ce mécanisme de livraison. Les fonctions de niveau supérieur de L2TP ne sont pas concernées par la retransmission ou le rangement des messages de contrôle. Le mécanisme de messagerie de contrôle fiable est un mécanisme de fenêtre glissante qui fournit un contrôle de la retransmission et de l'encombrement du message de contrôle. Chaque homologue entretient un état séparé de numéros de séquence pour chaque connexion de contrôle.

Le numéro de séquence de message, Ns, commence à 0. Chaque message suivant est envoyé avec le prochain incrément du numéro de séquence. Le numéro de séquence est donc un compteur de fonctionnement libre représenté modulo 65536. Le numéro de séquence dans l'en-tête d'un message reçu est considéré comme inférieur ou égal au dernier numéro reçu si sa valeur est dans la gamme du dernier numéro reçu et des 32767 valeurs précédentes, incluses. Par exemple, si le dernier numéro de séquence reçu était 15, les messages avec des numéros de séquence de 0 à 15, ainsi que de 32784 à 65535, seraient considérés comme inférieurs ou égaux. Un tel message serait considéré comme une duplication d'un message déjà reçu et ignoré pour le

traitement. Cependant, afin de s'assurer que tous les messages reçoivent un accusé de réception approprié (en particulier dans le cas d'un message ACK perdu) la réception de messages dupliqués DOIT être acquittée par le mécanisme de livraison fiable. Cet accusé de réception peut soit être porté en queue sur un message, soit envoyé explicitement via un message ACK.

Tous les messages de contrôle prélèvent un intervalle dans l'espace de numéros de séquence de message de contrôle, sauf le message ACK. Donc, Ns n'est pas incrémenté après l'envoi d'un message ACK.

Le dernier numéro de message reçu, Nr, est utilisé pour accuser réception des messages reçus par un homologue L2TP. Il contient le numéro de séquence du message que l'homologue s'attend à recevoir ensuite (par exemple, le dernier Ns d'un message non ACK reçu plus 1, modulo 65536). Alors que le Nr dans un message ACK reçu est utilisé pour purger les messages d'une file d'attente de retransmission locale (voir ci-dessous) le Nr du prochain message envoyé n'est pas mis à jour par le Ns du message ACK. La vérification de la bonne santé du Nr DEVRAIT être vérifiée avant de purger la file d'attente de retransmission. Par exemple, si le Nr reçu dans un message de contrôle est supérieur au dernier Ns envoyé plus 1 modulo 65536, le message de contrôle est clairement invalide.

Le mécanisme de livraison fiable chez un homologue receveur est chargé de s'assurer que les messages de contrôle sont livrés dans l'ordre et sans duplication au niveau supérieur. Les messages qui arrivent déclassés peuvent être mis en file d'attente afin d'être livrés lorsque les messages manquants seront reçus. Autrement, ils peuvent être éliminés, ce qui exige une retransmission par l'homologue. Lors de l'élimination des paquets de contrôle décalés, Nr PEUT être mis à jour avant que le paquet soit éliminé.

Chaque connexion de contrôle tient une file d'attente des messages de contrôle à transmettre à son homologue. Le message en tête de la file d'attente est envoyé avec une certaine valeur Ns et est conservé jusqu'à ce qu'un message de contrôle arrive de l'homologue dans lequel le champ Nr indique la réception de ce message. Après qu'un certain temps (la valeur par défaut recommandée est 1 seconde mais DEVRAIT être configurable) se passe sans accusé de réception, le message est retransmis. Le message retransmis contient la même valeur Ns, mais la valeur Nr DOIT être mise à jour avec le numéro de séquence du prochain message attendu.

Chaque retransmission suivante d'un message DOIT employer un intervalle de retard exponentiel. Donc, si la première retransmission s'est produite après une seconde, la retransmission suivante devrait intervenir après que deux secondes se sont écoulées, puis quatre secondes, etc. Une mise en œuvre PEUT placer un maximum à l'intervalle entre les retransmissions. Ce maximum NE DEVRAIT PAS être inférieur à huit secondes par retransmission. Si aucune réponse de l'homologue n'est détectée après plusieurs retransmissions (une valeur par défaut recommandée est de dix, mais DOIT être configurable) la connexion de contrôle et toutes les sessions associées DOIVENT être éliminées. Comme c'est le premier message à établir une connexion de contrôle, le SCCRQ PEUT employer un maximum de retransmissions différent des autres messages de contrôle afin d'aider à faciliter le repli à temps sur des LCCE de remplacement.

Lorsque une connexion de contrôle va être fermée pour des raisons autres que la perte de la connexité, les mécanismes d'état et livraison fiable DOIVENT être conservés en fonctionnement pendant l'intervalle de retransmission complet après l'envoi du message StopCCN final (par exemple, 1 + 2 + 4 + 8 + 8... secondes) ou jusqu'à ce que le message StopCCN ait été lui-même acquitté.

Un mécanisme de fenêtre glissante est utilisé pour la transmission et retransmission des messages de contrôle. Considérons deux homologues, A et B. Supposons que A spécifie une AVP Taille de la fenêtre de réception d'une valeur de N dans le message SCCRQ ou SCCRP. Il est maintenant permis à B d'avoir un maximum de N messages de contrôle en cours (c'est-à-dire, non acquittés). Une fois que N messages ont été envoyés, B doit attendre un accusé de réception de la part de A qui fasse avancer la fenêtre avant d'envoyer de nouveaux messages de contrôle. Une mise en œuvre peut annoncer une fenêtre de réception non à zéro aussi petite ou aussi grande qu'elle veut, selon sa propre capacité à traiter les messages entrants avant d'envoyer un accusé de réception. Chaque homologue DOIT limiter le nombre de messages non acquittés qu'il va envoyer avant de recevoir un accusé de réception à cette taille de la fenêtre de réception. La profondeur réelle de la file d'attente interne des messages non acquittés peut être encore limitée par l'allocation des ressources locales ou par des mécanismes dynamiques de démarrage lent et d'évitement d'encombrement.

Lors de la retransmission des messages de contrôle, une procédure d'ajustement de fenêtre de démarrage lent et d'évitement d'encombrement DEVRAIT être utilisée. Une procédure recommandée est décrite à l'Appendice A. Un homologue PEUT éliminer les messages, mais NE DOIT PAS retarder activement l'accusé de réception des messages comme technique de contrôle de flux des messages de contrôle. L'Appendice B contient des exemples de transmission de message de contrôle, d'accusé de réception, et de retransmission.

4.3 Authentification du message de contrôle

L2TP incorpore une vérification facultative d'authentification et d'intégrité pour tous les messages de contrôle. Ce mécanisme

consiste en un hachage unidirectionnel calculé sur l'en-tête et le corps du message de contrôle L2TP, un secret partagé préconfiguré, et un nom occasionnel local et distant (de valeur aléatoire) échangé via l'AVP Nom occasionnel d'authentification du message de contrôle. Cette vérification d'authentification et d'intégrité par message est conçue pour effectuer une authentification mutuelle entre les nœuds L2TP, effectuer la vérification d'intégrité de tous les messages de contrôle, et protéger contre la falsification des messages de contrôle et les attaques en répétition qui seraient autrement très faciles à monter.

Au moins un secret partagé (mot de passe) DOIT exister entre les nœuds L2TP communicants pour permettre l'authentification du message de contrôle. Voir au paragraphe 5.4.3 les détails du calcul du résumé de message et de la construction des AVP Nom occasionnel d'authentification et Résumé de message de message de contrôle.

L'authentification de message de contrôle L2TPv3 est similaire à l'authentification de tunnel L2TPv2 [RFC2661] par son utilisation d'un secret partagé et son calcul de hachage unidirectionnel. La principale différence est que au lieu de calculer le hachage sur le contenu choisi d'un message de contrôle reçu (par exemple, les AVP Challenge et Type de message) comme dans L2TPv2, le message entier est utilisé dans le hachage dans L2TPv3. De plus, au lieu d'inclure le résumé du hachage dans les seuls messages SCCRP et SCCCN, il est maintenant inclus dans tous les messages L2TP.

Le mécanisme d'authentification de message de contrôle est facultatif, et peut être désactivé si les deux homologues en sont d'accord. Par exemple, si IPsec est déjà utilisé pour les vérifications de sécurité et d'intégrité entre les LCCE, la fonction du mécanisme L2TP devient redondante et peut être désactivée.

La présence de l'AVP Nom occasionnel d'authentification de message de contrôle dans un message SCCRQ ou SCCRP sert d'indication pour l'homologue que l'authentification de message de contrôle est activée. Si un SCCRQ ou SCCRP contient une AVP Nom occasionnel d'authentification de message de contrôle, le receveur du message DOIT répondre par une AVP Résumé de message dans tous les messages suivants envoyés. L'authentification de message de contrôle est toujours bidirectionnelle ; soit les deux côtés participent à l'authentification, soit aucun des deux.

Si l'authentification de message de contrôle est désactivée, l'AVP Résumé de message PEUT quand même être envoyée comme vérification de l'intégrité du message. La vérification d'intégrité est calculée comme au paragraphe 5.4.3, avec un secret partagé vide de longueur zéro, un nom occasionnel local, et un nom occasionnel distant. Si un résumé de message invalide est reçu, on devrait supposer que le message a été corrompu dans le transit et le message devrait être éliminé en conséquence.

Les mises en œuvre PEUVENT limiter le débit des messages de contrôle, en particulier les messages SCCRQ, en réception pour des raisons de performances ou pour la protection contre des attaques de déni de service.

4.4 Maintien en vie (Hello)

L2TP emploie un mécanisme de maintien en vie pour détecter la perte de connexité entre une paire de LCCE. Ceci est accompli en injectant des messages de contrôle Hello (voir au paragraphe 6.5) après qu'un délai s'est écoulé depuis la réception du dernier message de données ou de contrôle sur respectivement une session ou une connexion de contrôle L2TP. Comme pour tout autre message de contrôle, si le message Hello n'est pas livré de façon fiable, le LCCE envoyeur déclare que la connexion de contrôle est en panne et rétablit l'état pour la connexion de contrôle. Ce comportement assure qu'une défaillance de la connexité entre les LCCE est détectée indépendamment par chaque extrémité d'une connexion de contrôle.

Comme le canal de contrôle fonctionne dans la bande avec le trafic de données sur le PSN, ce seul mécanisme peut être utilisé pour en déduire la connexité des données de base entre une paire de LCCE pour toutes les sessions associées à la connexion de contrôle.

Des "Maintien en vie" périodiques pour la connexion de contrôle DOIVENT être mis en œuvre par l'envoi d'un Hello si un certain délai (une valeur par défaut recommandée est de 60 secondes, mais DOIT être configurable) s'est passé sans recevoir de message (de données ou de contrôle) de l'homologue. Un LCCE qui envoie des messages Hello sur plusieurs connexions de contrôle entre les mêmes points d'extrémité de LCCE DOIT employer un mécanisme de temporisateur à gigue pour empêcher le groupement des messages Hello.

4.5 Transmission des trames de données de session

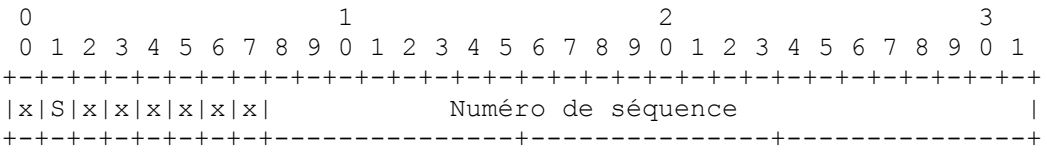
Une fois l'établissement de session achevé, des trames de circuit sont reçues à un LCCE, encapsulées dans L2TP (avec une attention appropriée au tramage, comme décrit dans les documents pour le type pseudo filaire particulier) et transmises sur la session appropriée. Pour chaque message de données sortant, l'envoyeur place l'identifiant spécifié dans l'AVP Identifiant de session local (reçu de l'homologue durant l'établissement de session) dans le champ Identifiant de session de l'en-tête de données L2TP. De cette manière, les trames de session sont multiplexées et démultiplexées entre une certaine paire de LCCE. Plusieurs connexions de contrôle peuvent exister entre une certaine paire de LCCE, et plusieurs sessions peuvent être associées à une certaine connexion de contrôle.

L'homologue LCCE qui reçoit le paquet de données L2TP identifie la session à laquelle est associé le paquet par l'identifiant de session dans l'en-tête du paquet de données. Le LCCE vérifie ensuite le champ Mouchard dans le paquet de données par rapport à la valeur du mouchard reçu dans l'AVP Mouchard allouée durant l'établissement de session. Il est important que les mises en œuvre notent que la vérification du champ Mouchard survient après la recherche du contexte de session par l'identifiant de session, et à ce titre, consiste simplement en une confrontation de la valeur du champ Mouchard avec celle mémorisée dans le contexte restitué. Il n'est pas besoin d'effectuer une recherche à travers l'identifiant de session et le mouchard comme une seule valeur. Tous les paquets de données reçus qui contiennent des identifiants de session ou des valeurs de mouchard associé invalides DOIVENT être éliminés. Finalement, le LCCE soit transmet le paquet réseau au sein de la trame tunnelée (par exemple, comme LNS) soit passe la trame à un circuit (par exemple, comme LAC).

4.6 Sous couche par défaut spécifique de couche 2

Le présent document définit un format par défaut de sous couche spécifique de couche 2 (voir au paragraphe 3.2.2) qu'un pseudo filaire peut utiliser pour des caractéristiques telles qu'une prise en charge séquentielle, un inter fonctionnement de couche 2, les opérations, administration et maintenance (OAM), ou d'autres opérations par paquet de données. La sous couche spécifique de couche 2 par défaut DEVRAIT être utilisée par un certain type pseudo filaire pour prendre en charge ces caractéristiques si c'est adéquat, et si sa présence est requise par un homologue durant la négociation de session. D'autres sous couches de remplacement PEUVENT être définies (par exemple, une encapsulation avec un plus grand champ Numéro de séquence ou des informations de temporisation) et identifiées pour être utilisées via l'AVP Type de sous couche spécifique de couche 2.

Figure 4.6 : Format de sous couche spécifique de couche 2 par défaut



Le bit S (Séquence) est réglé à 1 lorsque le numéro de séquence contient un numéro valide pour cette séquence de trame. Si le bit S est réglé à zéro, le contenu de Numéro de séquence est indéfini et DOIT être ignoré par le receveur.

Le champ Numéro de séquence contient un compteur de fonctionnement libre de 2^24 numéros de séquence. Si le numéro dans ce champ est valide, le bit S DOIT être réglé à 1. Le numéro de séquence commence à zéro, qui est un numéro de séquence valide. (De cette façon, les mises en œuvre qui insèrent des numéros de séquence n'ont pas à "sauter" le zéro en incrémentant.) Le numéro de séquence dans l'en-tête d'un message reçu est considéré comme inférieur ou égal au dernier numéro reçu si sa valeur se tient dans la gamme des derniers numéros reçus et des (2^23-1) valeurs précédentes, incluses.

4.6.1 Séquençage des paquets de données

Le champ Numéro de séquence peut être utilisé pour détecter la perte, la duplication, ou le décalage de paquets au sein c'une certaine session.

Lorsque des trames de couche 2 sont portées sur un canal de données L2TP sur IP ou L2TP sur UDP/IP, cette partie de la liaison a comme caractéristique d'être capable de réordonner, dupliquer, ou éliminer en silence les paquets. La remise en ordre peut rompre avec certains protocoles non IP ou trafics de contrôle de couche 2 qui sont portés par la liaison. L'élimination en silence ou la duplication de paquets peut rompre avec des protocoles qui supposent des indications d'erreur par paquet, comme la compression d'en-tête TCP. Bien qu'un mécanisme commun pour la détection de la suite des paquets soit fourni, les caractéristiques de dépendance à la séquence des protocoles individuels sortent du domaine d'application du présent document.

Si un protocole transporté sur des canaux de données L2TP ne peut pas tolérer le désordre des paquets de données, la duplication de paquet, ou la perte silencieuse de paquet, le séquençage peut être activé sur certains paquets ou tous les paquets en utilisant le bit S et le champ Numéro de séquence défini dans la sous couche par défaut spécifique de couche 2 (voir au

paragraphe 4.6). Pour une certaine session L2TP, chaque LCCE est responsable de communiquer à son homologue le niveau de prise en charge du séquençage qu'il exige des paquets de données qu'il reçoit. Les mécanismes pour annoncer des informations durant la négociation de session sont fournis (voir l'AVP Séquençage des données au paragraphe 5.4.4).

Pour déterminer si un paquet est en ou hors séquence, une mise en œuvre DEVRAIT utiliser une méthode résiliente aux pannes temporaires de connexité couplée avec de forts taux de paquets par session. La méthode recommandée est précisée à l'Appendice C.

4.7 Interopérabilité et migration de L2TPv2/v3

Les environnements L2TPv2 et L2TPv3 devraient être capables de coexister lors de la migration vers L2TPv3. Les questions de migration sont discutées pour chaque type de support dans cette section. La plupart des questions ne s'appliquent qu'aux mises en œuvre qui exigent le fonctionnement des deux L2TPv2 et L2TPv3.

Cependant, même les mises en œuvre de L2TPv3 seul doivent au moins être conscientes de ces questions afin d'interopérer avec les mises en œuvre qui prennent en charge les deux versions.

4.7.1 L2TPv3 sur IP

Les mises en œuvre L2TPv3 qui fonctionnent strictement sur IP sans désir d'interopérer avec les mises en œuvre L2TPv2 peuvent en toute sécurité laisser de côté toutes les questions de migration à partir de L2TPv2. Tous les messages de contrôle et les messages de données sont envoyés comme décrit dans le présent document, sans référence normative à la RFC2661.

Si on souhaite tunneler PPP sur L2TPv3, et ne revenir à L2TPv2 que si il n'est pas disponible, alors L2TPv3 sur UDP avec repli automatique (voir au paragraphe 4.7.3) DOIT être utilisé. Il n'y a pas de méthode déterministe de repli automatique de L2TPv3 sur IP à L2TPv2 ou L2TPv3 sur UDP. On pourrait en déduire si L2TPv3 sur IP est pris en charge en envoyant un SCCRQ et en attendant une réponse, mais ceci pourrait poser des problèmes durant des périodes de pertes de paquets entre les nœuds L2TP.

4.7.2 L2TPv3 sur UDP

Le format de l'en-tête L2TPv3 sur UDP est défini au paragraphe 4.1.2.1.

En fonctionnement sur UDP, L2TPv3 utilise le même accès (1701) que L2TPv2 et partage les deux premiers octets du format d'en-tête avec L2TPv2. Le champ Version est utilisé pour distinguer les paquets L2TPv2 des paquets L2TPv3. Si une mise en œuvre est capable de fonctionner en mode L2TPv2 ou L2TPv3, il est possible de détecter automatiquement si un homologue peut prendre en charge L2TPv2 ou L2TPv3 et fonctionner en conséquence. Les détails de cette capacité de repli sont définis dans les paragraphes qui suivent.

4.7.3 Repli automatique sur L2TPv2

Quand elle fonctionne sur UDP, une mise en œuvre peut détecter si un homologue est à capacité L2TPv3 en envoyant un SCCRQ spécial qui est proprement formaté pour L2TPv2 et L2TPv3. Ceci se fait en envoyant un SCCRQ avec son champ Version réglé à 2 (pour L2TPv2) et en s'assurant que toutes les AVP spécifiques de L2TPv3 (c'est-à-dire, les AVP présentes au sein du présent document et non définies dans la RFC2661) dans le message sont envoyées avec chaque bit M réglé à 0, et que toutes les AVP L2TPv2 sont présentes comme elles le seraient pour L2TPv2. On fait cela pour que les AVP L2TPv3 soient ignorées par une mise en œuvre seulement L2TPv2. Noter que, dans L2TPv2 et L2TPv3, la valeur contenue dans l'espace de l'en-tête du message de contrôle utilisé par l'identifiant de connexion de contrôle de 32 bits dans L2TPv3, et l'identifiant de tunnel de 16 bits et l'identifiant de session de 16 bits dans L2TPv2, sont toujours à 0 pour un SCCRQ. Cela cache effectivement le fait qu'il y a une paire de champs de 16 bits dans L2TPv2, et un seul champ de 32 bits dans L2TPv3.

Si la mise en œuvre homologue est à capacité L2TPv3, un message de contrôle avec le champ Version réglé à 3 et un en-tête et le format de message L2TPv3 sera envoyé en réponse au SCCRQ. Le fonctionnement peut alors se poursuivre comme L2TPv3. Si un message est reçu avec le champ Version réglé à 2, on doit supposer que la mise en œuvre homologue est seulement L2TPv2, activant donc le repli sur le mode L2TPv2 de façon sûre.

Note : le mode d'auto détection L2TPv2/v3 exige que toutes les mises en œuvre L2TPv3 sur UDP soient libérales en acceptant un message de contrôle SCCRQ avec le champ Version réglé à 2 ou 3 et la présence d'AVP spécifiques de L2TPv2. Une mise en œuvre seulement L2TPv3 DOIT ignorer toutes les AVP L2TPv2 (par exemple, celles définies dans la RFC2661 et qui ne sont pas dans le présent document) au sein d'un SCCRQ avec le champ Version réglé à 2 (même si le bit M est établi sur les AVP spécifiques de L2TPv2).

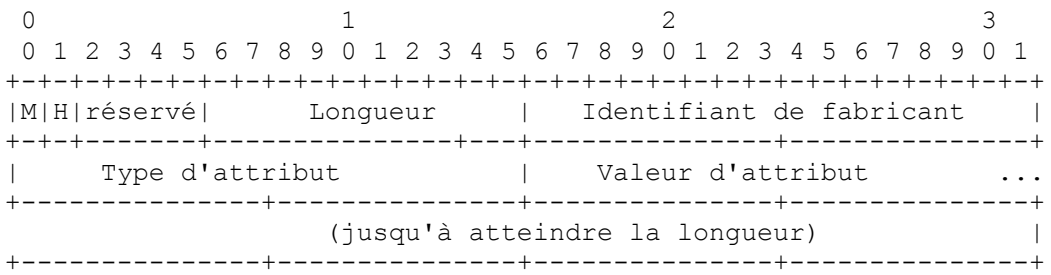
5. Paires d’attribut/valeur de message de contrôle

Pour maximiser l'extensibilité tout en permettant l'interopérabilité, une méthode uniforme de codages des types de message est utilisée dans L2TP. Ce codage sera appelé une paire de valeur d'attribut (AVP, *Attribute Value Pair*) dans la suite de ce document.

5.1 Format d’une AVP

Chaque AVP est codée comme suit :

Figure 5.1 : Format d’AVP



Les six premiers bits comprennent un gabarit binaire qui décrit les attributs généraux de l'AVP. Deux bits sont définis dans le présent document ; les bits restants sont réservés pour des extensions futures. Les bits réservés DOIVENT être réglés à 0 à l'envoi et ignorés à réception.

Bit M (Obligatoire) : il contrôle le comportement exigé d'une mise en œuvre qui reçoit une AVP non reconnue. Le bit M d'une certaine AVP DOIT être inspecté et faire l'objet d'une action seulement si l'AVP n'est pas reconnue (voir au paragraphe 5.2).

Bit caché (H, *Hidden*) : il identifie la dissimulation des données dans le champ Valeur d'attribut d'une AVP. Cette capacité peut être utilisée pour éviter de passer des données sensibles, comme un mot de passe d'utilisateur, en clair dans une AVP. Le paragraphe 5.3 décrit la procédure pour effectuer la dissimulation d'une AVP.

Longueur : elle contient le nombre d'octets (incluant les champs Longueur globale et Gabarit binaire) contenus dans cette AVP. La longueur peut être calculée comme 6 + longueur du champ Valeur d'attribut en octets. Le champ lui-même est de 10 bits, permettant un maximum de 1023 octets de données dans une seule AVP. La longueur minimum d'une AVP est 6. Si la longueur est 6, le champ Valeur d'attribut est alors absent.

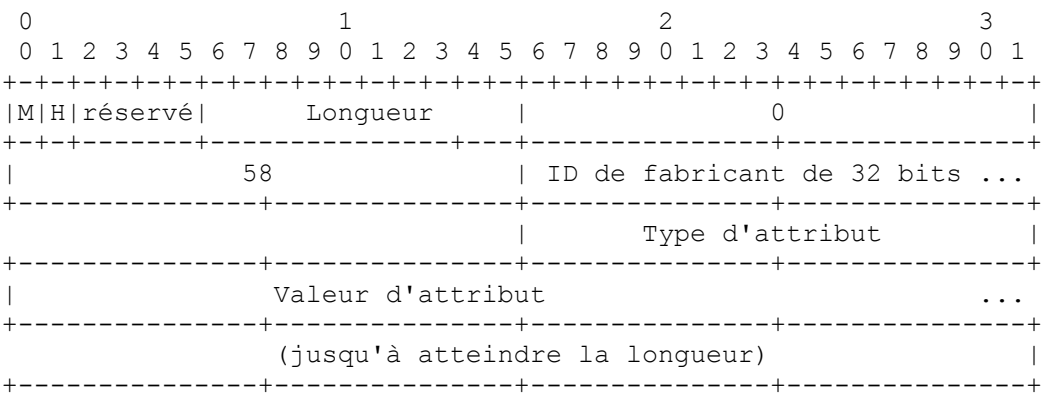
Identifiant de fabricant : c'est la valeur du "code d'entreprise privée de gestion de réseau SMI" [RFC1700] alloué par l'IANA. La valeur 0, correspondant aux valeurs d'attribut adoptées par l'IETF, est utilisée pour toutes les AVP définies dans le présent document. Tout fabricant qui souhaite mettre en œuvre ses propres extensions L2TP peut utiliser son propre identifiant de fabricant avec ses valeurs d'attribut privées, garantissant qu'il ne va pas entrer en collision avec les extensions d'un autre fabricant ou de futures extensions de l'IETF. Noter qu'il y a 16 bits alloués pour l'identifiant de fabricant, limitant donc cette caractéristique aux 65 535 premières entreprises.

Type d'attribut : valeur de deux octets avec une interprétation unique sur toutes les AVP définies sous un certain ID de fabricant.

Valeur d'attribut : c'est la valeur réelle comme indiqué par l'identifiant de fabricant et le type d'attribut. Il suit immédiatement après le champ Type d'attribut et couvre les octets restants indiqués dans le champ Longueur (c’est-à-dire, Longueur moins 6 octets d'en-tête). Ce champ est absent si la longueur est 6.

Dans le cas où l'espace d'identifiant de fabricant de 16 bits serait épuisé, des AVP spécifiques de fabricant avec un identifiant de fabricant de 32 bits DOIVENT être encapsulées de la manière suivante :

Figure 5.2 : Format d'AVP d'identifiant étendu de fabricant



Cette AVP code une AVP spécifique de fabricant avec un espace d'identifiant de fabricant de 32 bits au sein du champ Valeur d'attribut. Plusieurs AVP de ce type peuvent exister dans tout message. L'identifiant de fabricant de 16 bits DOIT être 0, qui indique que c'est une AVP définie par l'IETF, et le Type d'attribut DOIT être 58, qui indique que ce qui suit est une AVP spécifique de fabricant avec un code d'identifiant de fabricant de 32 bits. Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DOIT être réglé à 0. La longueur de l'AVP est 12 plus la longueur de la valeur d'attribut.

5.2. AVP obligatoires et réglage du bit M

Si le bit M est établi sur une AVP qui n'est pas reconnue par son receveur, la session ou la connexion de contrôle associée au message de contrôle qui contient l'AVP DOIT être fermée. Si le message de contrôle qui contient l'AVP non reconnue est associé à une session (par exemple, une ICRQ, ICRP, ICCN, SLI, etc.) alors la session DOIT recevoir une CDN avec un code de résultat de 2 et un code d'erreur de 8 (comme défini au paragraphe 5.4.2) et être fermée. Si le message de contrôle qui contient l'AVP non reconnue est associé à un établissement ou une maintenance d'une connexion de contrôle (par exemple, SCCRQ, SCCRP, SCCCN, Hello) alors la connexion de contrôle associée DOIT recevoir un StopCCN avec un code de résultat de 2 et un code d'erreur de 8 (comme défini au paragraphe 5.4.2) et être fermée. Si le bit M n'est pas établi sur une AVP non reconnue, l'AVP DOIT être ignorée à réception, en traitant le message de contrôle comme si l'AVP n'était pas présente.

La réception d'une AVP non reconnue qui a le bit M établi est catastrophique pour la session ou connexion de contrôle à laquelle elle est associée. Donc, le bit M ne devrait être établi que pour les AVP qui sont réputées cruciales pour le bon fonctionnement de la session ou connexion de contrôle par l'envoyeur. Les AVP qui sont considérées comme cruciales par l'envoyeur peuvent varier selon l'application et les options configurées. En aucune cas un receveur d'une AVP ne devra la "valider" si le bit M est établi sur une AVP reconnue. Si l'AVP est reconnue (comme toutes les AVP définies dans le présent document DOIVENT l'être pour une spécification L2TPv3 conforme) alors par définition, le bit M est sans conséquence.

L'envoyeur d'une AVP est libre d'établir son bit M à 1 ou à 0 selon que l'application configurée exige strictement que la valeur contenue dans l'AVP soit reconnue ou non. Par exemple, le "repli automatique sur L2TPv2" du paragraphe 4.7.3 exige que le bit M soit réglé sur toutes les nouvelles AVP L2TPv3 à zéro si le repli sur L2TPv2 est pris en charge et désiré, et à 1 sinon.

Le bit M est utile comme assurance supplémentaire de prise en charge des AVP d'extensions critiques. Cependant, des méthodes plus explicites peuvent être disponibles pour déterminer la prise en charge d'une certaine caractéristique plutôt que d'utiliser le bit M seul. Par exemple, si une nouvelle AVP est définie dans un message pour lequel il y a toujours un message de réponse (c'est-à-dire, un message ICRQ, ICRP, SCCRQ, ou SCCRP) plutôt que d'envoyer simplement une AVP dans le message avec le bit M établi, la disponibilité de l'extension peut être identifiée par l'envoi d'une AVP dans le message de demande et d'attendre une AVP correspondante dans le message de réponse. Cette méthode plus explicite est préférée, lorsque possible.

Le bit M joue aussi un rôle pour déterminer si une valeur malformée ou hors gamme au sein d'une AVP devrait être ignorée ou devrait résulter en la terminaison d'une session ou connexion de contrôle (voir les détails au paragraphe 7.1).

5.3 Dissimulation de la valeur et de l'attribut de l'AVP

Le bit H dans l'en-tête de chaque AVP fournit un mécanisme pour indiquer à l'homologue receveur si le contenu de l'AVP est caché ou présent en clair. Ce dispositif peut être utilisé pour cacher des données sensibles d'un message de contrôle comme les mots de passe d'usagers, les identifiants, ou d'autres informations vitales.

Le bit H ne DOIT être établi que si (1) un secret partagé existe entre les LCCE et (2) le message de contrôle Authentification

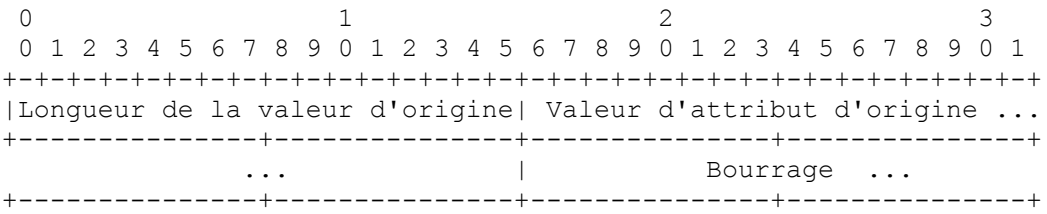
est activé (voir au paragraphe 4.3). Si le bit H est établi dans une ou des AVP dans un certain message de contrôle, au moins une AVP Vecteur aléatoire doit aussi être présente dans le message et DOIT précéder la première AVP qui a un bit H de 1.

Le secret partagé entre les LCCE est utilisé pour déduire une clé partagée unique pour cacher et divulguer des calculs. La clé partagée déduite est obtenue via un hachage chiffré HMAC-MD5 [RFC2104], dont la clé consiste en le secret partagé, et les données hachées consistent en un seul octet contenant la valeur 1.

clé_partagée = HMAC_MD5 (secret_partagé, 1)

Cacher une valeur d'AVP se fait en plusieurs étapes. La première est de prendre les champs Longueur et Valeur de l'AVP d'origine (en clair) et de les coder dans le sous format d'AVP cachée, qui apparaît comme suit :

Figure 5.3 : Sous format d'AVP cachée



Longueur de la valeur d'attribut d'origine : c'est la longueur de la valeur originale de l'attribut à cacher en octets. Ceci est nécessaire pour déterminer la longueur d'origine de la valeur de l'attribut qui est perdue lorsque le bourrage supplémentaire est ajouté.

Valeur d'attribut d'origine : c'est la valeur de l'attribut qui est obscurci.

Bourrage : octets aléatoires supplémentaires utilisés pour obscurcir la longueur de la valeur de l'attribut qui est à cacher.

Pour masquer la taille des données qu'on cache, le sous format résultant PEUT être bourré comme montré ci-dessus. Le bourrage N'ALTÈRE PAS la valeur placée dans le champ Longueur de la valeur de l'attribut d'origine, mais altère la longueur de l'AVP résultante créée. Par exemple, si une valeur d'attribut à cacher est de quatre octets, la longueur non cachée d'AVP serait de 10 octets (6 + longueur de la valeur de l'attribut). Après être cachée, la longueur de l'AVP va devenir 6 + longueur de la valeur d'attribut + taille du champ Longueur de la valeur de l'attribut d'origine + Bourrage. Donc, si Bourrage fait 12 octets, la longueur de l'AVP sera 6 + 4 + 2 + 12 = 24 octets.

Ensuite, un hachage MD5 [RFC1321] est effectué (dans l'ordre des octets du réseau) sur l'enchaînement de ce qui suit :
+ le nombre d'attribut de deux octets de l'AVP + la clé partagée + une valeur aléatoire de longueur arbitraire

La valeur aléatoire utilisée dans ce hachage est passée dans le champ Valeur d'une AVP Vecteur aléatoire. Cette AVP doit être placée dans le message par l'envoyeur avant toute AVP cachée. La même valeur aléatoire peut être utilisée pour plus d'une AVP cachée dans le même message, mais pas pour cacher deux instances ou plus d'une AVP avec le même type d'attribut sauf si les valeurs d'attribut sont aussi identiques dans les deux AVP. Lorsque une valeur aléatoire différente est utilisée pour cacher des AVP successives, une nouvelle AVP Vecteur aléatoire DOIT être placée dans le message de contrôle avant la première AVP à laquelle elle s'applique.

La valeur du hachage MD5 est alors OUxée avec le premier segment de 16 octets (ou moins) du sous format d'AVP cachée et placée dans le champ valeur d'attribut de l'AVP cachée. Si le sous format d'AVP cachée fait moins de 16 octets, le sous format est transformé comme si le champ Valeur d'attribut avait été bourré à 16 octets avant le OUx. Seuls les octets réellement présents dans le sous format sont modifiés, et la longueur de l'AVP n'est pas altérée.

Si le sous format fait plus de 16 octets, un second hachage unidirectionnel MD5 est calculé sur un flux d'octets consistant en la clé partagée suivie par le résultat du premier OUx. Ce hachage est OUxé avec le second segment de 16 octets (ou moins) du sous format et placé dans les octets correspondants du champ Valeur de l'AVP cachée.

Si nécessaire, cette opération est répétée, en utilisant la clé partagée avec chaque résultat de OUx pour générer le prochain hachage afin de OUxer avec lui le prochain segment de la valeur.

La méthode de dissimulation a été adaptée de la [RFC2865], qui a été tirée du chapitre "Mixing in the Plaintext" dans l'ouvrage "Network Security" de Kaufman, Perlman et Speciner [KPS]. Voici une explication détaillée de la méthode :

On appelle S la clé partagée, RV la valeur aléatoire, et A le type d'attribut. On fractionne le champ de valeur de tronçons de

16 octets p_1, p_2, etc., dont le dernier est bourré à la fin de données aléatoires jusqu'à une limite de 16 octets. On appelle c_1, c_2, etc. les blocs de texte chiffré. On définit aussi des valeurs intermédiaires b_1, b_2, etc.

```
b_1 = MD5 (A + S + RV)   c_1 = p_1 OUx b_1
b_2 = MD5 (S + c_1)      c_2 = p_2 OUx b_2
        .                .
        .                .
        .                .
b_i = MD5 (S + c_i-1)    c_i = p_i OUx b_i
```

La chaîne va contenir c_1 + c_2 +...+ c_i, où "+" note l'enchaînement.

À réception, la valeur aléatoire est prise de l'AVP Vecteur aléatoire rencontrée dans le message avant que l'AVP ne soit dévoilée. Le processus ci-dessus est inversé pour donner la valeur originale.

5.4 Sommaire des AVP

Les paragraphes qui suivent contiennent la liste de toutes les AVP L2TP définies dans le présent document.

À la suite du nom de l'AVP figure une liste qui indique les types de message qui utilisent chaque AVP. Après chaque titre d'AVP figure une brève description de l'objet de l'AVP, le détail (incluant un graphique) du format de la valeur d'attribut, et toutes les informations supplémentaires nécessaires pour le bon usage de l'AVP.

5.4.1 AVP générale de message de contrôle

Type de message (tous les messages)
L'AVP Type de message, type d'attribut 0, identifie le message de contrôle et définit le contexte dans lequel la signification exacte des AVP suivantes sera déterminée.

Le champ Valeur d'attribut pour cette AVP a le format suivant :

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Type de message          |
+---------------+---------------+
```

Le Type de message est un entier non signé de deux octets.

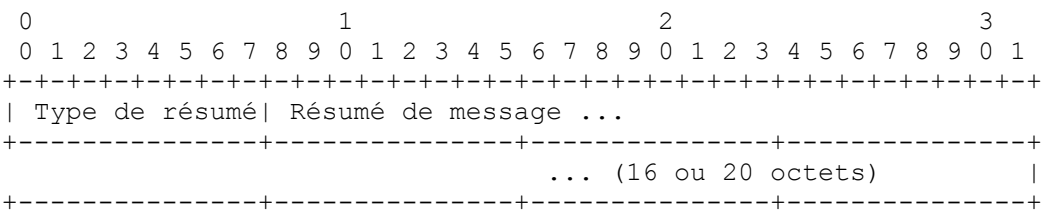
L'AVP Type de message DOIT être la première AVP d'un message, suivie immédiatement par l'en-tête de message de contrôle (défini au paragraphe 3.2.1). Voir au paragraphe 3.1 la liste des types de message de contrôle définis et leurs identifiants.

Le bit Obligatoire (M, *Mandatory*) au sein de l'AVP Type de message a une signification spéciale. Plutôt qu'une indication de si l'AVP elle-même devrait être ignorée si elle n'est pas reconnue, c'est une indication sur si le message de contrôle lui-même devrait être ignoré. Si le bit M est établi au sein de l'AVP Type de message et si le type de message est inconnu de la mise en œuvre, la connexion de contrôle DOIT être libérée. Si le bit M n'est pas établi, alors la mise en œuvre peut ignorer un type de message inconnu. Le bit M DOIT être établi à 1 pour tous les types de message définis dans le présent document. Cette AVP NE DOIT PAS être cachée (le bit H DOIT être 0). La longueur de cette AVP est 8.

Un message de contrôle spécifique de fabricant peut être défini en réglant le Vendor ID de l'AVP Type de message à une valeur autre que le Vendor ID IETF de 0 (voir au paragraphe 5.1). L'AVP Type de message DOIT toujours être la première AVP dans le message de contrôle.

Résumé de message (tous les messages)
L'AVP Résumé de message, type d'attribut 59, est utilisée comme vérification d'intégrité et d'authentification de l'en-tête et du corps du message de contrôle L2TP.

Le champ Valeur d'attribut pour cette AVP a le format suivant :



Type de résumé est un entier d'un octet qui indique l'algorithme de calcul de résumé :

0 : HMAC-MD5 [RFC2104]
1 : HMAC-SHA-1 [RFC2104]

Type de résumé 0 (HMAC-MD5) DOIT être pris en charge, tandis que Type de résumé 1 (HMAC-SHA-1) DEVRAIT être pris en charge.

Le résumé de message est de longueur variable et contient le résultat du calcul de l'authentification et de l'intégrité du message de contrôle. Pour Type de résumé 0 (HMAC-MD5), la longueur du résumé DOIT être 16 octets. Pour Type de résumé 1 (HMAC-SHA-1) la longueur du résumé DOIT être 20 octets.

Si l'authentification du message de contrôle est activée, au moins une AVP Résumé de message DOIT être présente dans tous les messages et DOIT être placée immédiatement après l'AVP Type de message. Cela force l'AVP Résumé de message à commencer avec un décalage bien connu et fixe. Une seconde AVP Résumé de message PEUT être présente dans un message et DOIT être placée directement après la première AVP Résumé de message.

Le secret partagé entre les LCCE est utilisé pour déduire une unique clé partagée pour le calcul de l'authentification du message de contrôle. La clé partagée déduite est obtenue via un hachage HMAC-MD5 chiffré selon la [RFC2104], dont la clé consiste en le secret partagé, et dont les données hachées consistent en un seul octet contenant la valeur 2.

clé_partagée = HMAC_MD5 (secret_partagé, 2)

Le calcul du résumé de message est comme suit pour tous les messages autres que le SCCRQ (où "+" marque l'enchaînement) :

Résumé de message = HMAC_Hash (shared_key, local_nonce + remote_nonce + control_message)

HMAC_Hash : Algorithme de hachage HMAC identifié par le type de résumé (MD5 ou SHA1)
local_nonce : Nom occasionnel choisi localement et annoncé au LCCE distant.
remote_nonce : Nom occasionnel reçu du LCCE distant (local_nonce et remote_nonce sont annoncés via l'AVP Nom occasionnel d'authentification de message de contrôle, qui est aussi définie dans cette section.)
shared_key : Clé partagée déduite pour cette connexion de contrôle.
control_message : C'est le contenu entier du message de contrôle L2TP, incluant l'en-tête de message de contrôle et toutes les AVP. Noter que l'en-tête de message de contrôle commence dans ce cas après l'identifiant de session tout à zéro quand on fonctionne sur IP (voir au paragraphe 4.1.1.2), et après l'en-tête UDP lorsque on fonctionne sur UDP (voir au paragraphe 4.1.2.1).

Pour le calcul du résumé de message, l'AVP Résumé de message DOIT être présente au sein du message de contrôle avec le type de résumé réglé à la valeur appropriée, mais le résumé de message lui-même est tout à zéro.

À la réception d'un message de contrôle, le contenu de l'AVP Résumé de message DOIT être comparé à la valeur de résumé attendue sur la base du calcul local. On fait cela en effectuant le même calcul de résumé que ci-dessus, en intervertissant le local_nonce et le remote_nonce. Cette vérification d'authenticité et d'intégrité de message DOIT être effectuée avant d'utiliser aucune information contenue dans le message de contrôle. Si le calcul échoue, le message DOIT être éliminé.

Le SCCRQ a droit à un traitement particulier car c'est le message initial qui commence une nouvelle connexion de contrôle. À ce titre, un seul nom occasionnel est disponible. Comme le nom occasionnel est présent dans le message lui-même au titre de l'AVP Nom occasionnel d'authentification de message de contrôle, il n'est pas nécessaire de l'utiliser explicitement dans le calcul. Le calcul du résumé de message SCCRQ est effectué comme suit :

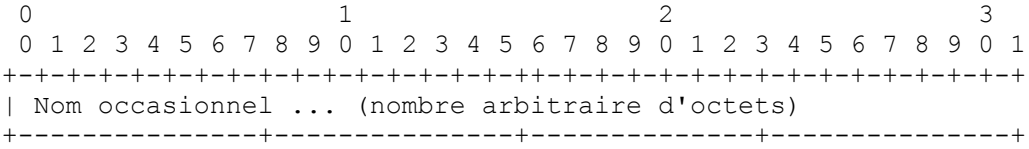
Résumé de message = HMAC_Hash (shared_key, control_message)

Pour permettre le passage en douceur à un nouveau secret partagé ou algorithme de hachage, deux AVP Résumé de message PEUVENT être présentes dans un message de contrôle, et deux secrets partagés PEUVENT être configurés pour un LCCE donné. Si deux AVP Résumé de message sont reçues dans un message de contrôle, le message DOIT être accepté si l'un ou

l'autre résumé de message est valide. Si deux secrets partagés sont configurés, chacun (séparément) DOIT être utilisé pour calculer un résumé à comparer au ou aux résumés de message reçus. Pour calculer un résumé pour un message de contrôle, le champ Valeur pour les deux AVP Résumé de message DOIT être réglé à zéro.

Cette AVP NE DOIT PAS être cachée (le bit H DOIT être 0). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur est 23 pour Type de résumé 1 (HMAC-MD5), et 27 pour Type de résumé 2 (HMAC-SHA-1).

Nom occasionnel d'authentification de message de contrôle (SCCRQ, SCCRP)
L'AVP Nom occasionnel d'authentification de message de contrôle, type d'attribut 73, DOIT contenir une valeur chiffrée aléatoire [RFC1750]. Cette valeur est utilisée pour l'authentification du message de contrôle. Le champ Valeur d'attribut pour cette AVP a le format suivant :

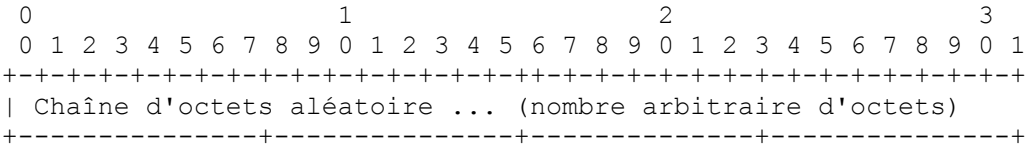


Le nom occasionnel est de longueur arbitraire, mais on recommande au moins 16 octets. Le nom occasionnel contient la valeur aléatoire à utiliser dans le calcul du hachage d'authentification du message de contrôle (voir la définition de l'AVP Résumé de message au début de ce paragraphe).

Si l'authentification de message de contrôle est activée, cette AVP DOIT être présente dans les messages SCCRQ et SCCRP.

Cette AVP NE DOIT PAS être cachée (le bit H DOIT être 0). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur de cette AVP est 6 plus la longueur du nom occasionnel.

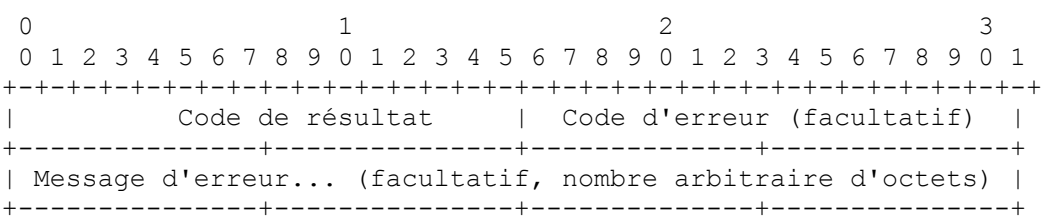
Vecteur aléatoire (tous les messages)
L'AVP Vecteur aléatoire, type d'attribut 36, DOIT contenir une valeur chiffrée aléatoire [RFC1750]. Cette valeur est utilisée pour cacher l'AVP. Le champ Valeur d'attribut pour cette AVP a le format suivant :



La chaîne d'octets aléatoire est de longueur arbitraire, mais on recommande au moins 16 octets. La chaîne contient le vecteur aléatoire à utiliser pour calculer le hachage MD5 pour restituer ou cacher la valeur d'attribut d'une AVP cachée (voir au paragraphe 5.3). Plus d'une AVP Vecteur aléatoire peuvent apparaître dans un message, dans lequel une AVP cachée utilise l'AVP Vecteur aléatoire précédente la plus proche. À ce titre, au moins une AVP Vecteur aléatoire DOIT précéder la première AVP avec le bit H établi. Cette AVP NE DOIT PAS être cachée (le bit H DOIT être à 0). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur de cette AVP est de 6 plus la longueur de la chaîne d'octets aléatoire.

5.4.2 Codes de résultat et d'erreur

Code de résultat (StopCCN, CDN)
L'AVP Code de résultat, type d'attribut 1, indique la raison de la fin de la connexion de contrôle ou session. Le champ Valeur d'attribut pour cette AVP a le format suivant :



Le code de résultat est un entier non signé de deux octets. Le code d'erreur facultatif est un entier non signé de deux octets. Un message d'erreur facultatif peut suivre le champ Code d'erreur. La présence du code et message d'erreur est indiquée par le champ Longueur d'AVP. Le message d'erreur contient une chaîne arbitraire qui fournit du texte supplémentaire (lisible par

l'homme) associé à la condition. Le texte lisible par l'homme dans tous les messages d'erreur DOIT être fourni dans le jeu de caractères UTF-8 [RFC3629] en utilisant le langage par défaut [RFC2277].

Cette AVP NE DOIT PAS être cachée (le bit H DOIT être 0). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur est 8 si il n'y a pas de code ou message d'erreur, 10 si il y a un code d'erreur et pas de message d'erreur, ou 10 plus la longueur du message d'erreur si il y a un code et un message d'erreur.

Les valeurs définies de code de résultat pour le message StopCCN sont comme suit :
0 - Réservé.
1 – Demande générale de libération de la connexion de contrôle.
2 – Erreur générale, le code d'erreur indique le problème.
3 – La connexion de contrôle existe déjà.
4 – Le demandeur n'est pas autorisé à établir une connexion de contrôle.
5 – La version de protocole du demandeur n'est pas acceptée, le code d'erreur indique la plus haute version supportée.
6 – Le demandeur est en train de fermer.
7 – Erreur d'automate à états finis ou fin de temporisation.

Les valeurs de code de résultat général pour le message CDN sont comme suit :
0 - Réservé.
1 - Session déconnectée par suite de perte de la porteuse ou de la déconnexion du circuit.
2 - Session déconnectée pour la raison indiquée dans le code d'erreur.
3 - Session déconnectée pour des raisons administratives.
4 – L'établissement de session a échoué parce que les facilités appropriées ne sont pas disponibles (condition temporaire).
5 - L'établissement de session a échoué parce que les facilités appropriées ne sont pas disponibles (condition permanente).
13 - Session non établie à cause de la perte du départage.
14 - Session non établie parce que le type de pseudo filaire n'est pas accepté.
15 - Session non établie, séquençage exigé sans sous couche spécifique de couche 2 valide.
16 – Erreur de l'automate à états finis ou fin de temporisation.
Des codes de résultat supplémentaires spécifiques du service sont définis en dehors du présent document.

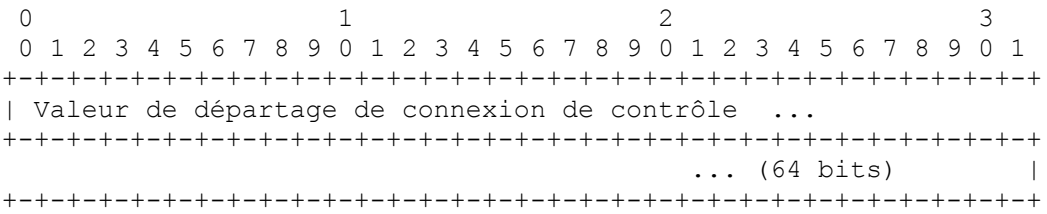
Les codes d'erreur définis ci-dessous relèvent de types d'erreurs qui ne sont pas spécifiques d'une demande L2TP particulière, mais plutôt des erreurs de protocole ou de format de message. Si une réponse L2TP indique dans son code de résultat qu'une erreur générale s'est produite, la valeur d'erreur générale devrait être examinée pour déterminer qu'elle était l'erreur. Les codes d'erreurs générales actuellement définis et leur signification sont comme suit :
0 – Pas d'erreur générale.
1 – Il n'existe pas encore de connexion de contrôle pour cette paire de LCCE.
2 – La longueur est fausse.
3 – Une des valeurs du champ est hors gamme.
4 – Ressources insuffisantes pour traiter maintenant cette opération.
5 – Identifiant de session invalide.
6 – Une erreur générique spécifique du fabricant s'est produite.
7 – Essayer un autre. Si l'initiateur a connaissance des autres destinations possibles de répondeur, il devrait essayer l'une d'elles. Ceci peut être utilisé pour guider un LAC ou LNS sur la base d'une politique.
8 – La session ou connexion de contrôle a été fermée suite à la réception d'une AVP inconnue avec le bit M établi (voir au paragraphe 5.2). Le message d'erreur DEVRAIT contenir l'attribut de l'AVP en cause sous forme (lisible par l'homme) textuelle.
9 – Essayer un autre dirigé. Si un LAC ou LNS a connaissance des autres destinations possibles, il devrait informer l'initiateur de la connexion ou session de contrôle. Le message d'erreur DOIT contenir une liste séparée par des virgules des adresses entre lesquelles l'initiateur peut choisir. Si le canal de données L2TP fonctionne sur IPv4, ce sera alors une liste séparée par des virgules des adresses IPv4 en format canonique décimal séparé par des points (par exemple, "192.0.2.1, 192.0.2.2, 192.0.2.3") dans le jeu de caractères UTF-8 [RFC3629] en utilisant le langage par défaut [RFC2277]. Si il n'y a pas de serveur à suggérer pour le LAC ou LNS, le code d'erreur 7 devrait alors être utilisé. Pour IPv4, le délimiteur entre les adresses DOIT être précisément une seule virgule et une seule espace. Pour IPv6, chaque adresse littérale DOIT être enclose entre des caractères "[" et "]", selon le codage décrit dans la [RFC2732].

Lorsque on utilise un code d'erreur général de 6, des informations supplémentaires sur l'erreur DEVRAIENT être incluses dans le champ Message d'erreur. Une AVP spécifique de fabricant PEUT être envoyée pour détailler plus précisément un problème spécifique du fabricant.

5.4.3 AVP de gestion de la connexion de contrôle

Départage de connexion de contrôle (SCCRQ)

L'AVP Départage de connexion de contrôle, type d'attribut 5, indique que l'envoyeur désire qu'il existe une seule connexion de contrôle entre une certaine paire de LCCE. Le champ Valeur d'attribut pour cette AVP a le format suivant :



La valeur du départage de connexion de contrôle est une valeur aléatoire de 8 octets qui est utilisée pour choisir une seule connexion de contrôle lorsque deux LCCE demandent concurremment une connexion de contrôle. Le receveur d'une SCCRQ doit vérifier pour voir si une SCCRQ a été envoyée à l'homologue ; si oui, un nœud a été détecté. Dans ce cas, le LCCE doit comparer sa valeur de départage de connexion de contrôle à celle reçue dans la SCCRQ. La plus faible valeur "gagne", et le "perdant" DOIT éliminer sa connexion de contrôle. Un StopCCN DEVRAIT être envoyé par le gagnant comme rejet explicite pour la SCCRQ perdante. Dans le cas où un départage est présent sur les deux côtés et où la valeur est égale, les deux côtés DOIVENT éliminer leur connexion de contrôles et recommencer la négociation de connexion de contrôle avec une nouvelle valeur aléatoire de départage.

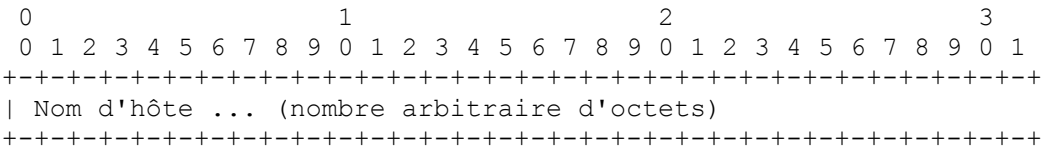
Si un départage est reçu et qu'une SCCRQ en instance n'a pas de valeur de départage, l'initiateur qui a inclus l'AVP Départage de connexion de contrôle "gagne". Si ni l'un ni l'autre n'a produit de départage, deux connexions de contrôle séparées sont ouvertes.

Les applications qui emploient un initiateur distinct et bien connu n'ont pas besoin de départage, et PEUVENT omettre cette AVP ou désactiver la fonction de départage. Les applications qui exigent le départage exigent aussi qu'un LCCE soit identifiable de façon univoque à réception d'une SCCRQ. Pour L2TP sur IP, ceci DOIT être accompli via l'AVP Identifiant de routeur.

Noter que dans la [RFC2661], cette AVP est appelée AVP "Départage" et n'est applicable qu'à une connexion de contrôle. Dans L2TPv3, l'AVP sert au même objet de départage, mais elle est applicable à une connexion de contrôle ou une session. L'AVP Départage de connexion de contrôle (présente seulement dans les messages de connexion de contrôle) et l'AVP Départage de session (présente seulement dans les messages de session) sont décrites séparément dans le présent document, mais partagent le même type d'attribut de 5.

Cette AVP NE DOIT PAS être cachée (le bit H DOIT être 0). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur de cette AVP est 14.

Nom d'hôte (SCCRQ, SCCRP)
L'AVP Nom d'hôte, type d'attribut 7, indique le nom du LAC ou LNS producteur, codé dans le jeu de caractères US-ASCII. Le champ Valeur d'attribut pour cette AVP a le format suivant :



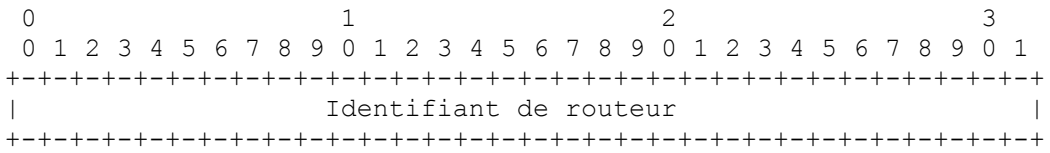
Le Nom d'hôte est de longueur arbitraire, mais DOIT être d'au moins 1 octet.

Ce nom devrait être aussi largement unique que possible ; pour les hôtes qui participent au DNS [RFC1034], un nom d'hôte avec un domaine pleinement qualifié serait approprié. Les AVP Nom d'hôte et/ou Identifiant de routeur DOIVENT être utilisées pour identifier un LCCE comme décrit au paragraphe 3.3.

Cette AVP NE DOIT PAS être cachée (le bit H DOIT être 0). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur de cette AVP est 6 plus la longueur de Nom d'hôte.

Identifiant de routeur (SCCRQ, SCCRP)
L'AVP Identifiant de routeur, type d'attribut 60, est un identifiant utilisé pour identifier un LCCE pour l'établissement de la connexion de contrôle, le départage, et/ou l'authentification de tunnel.

Le champ Valeur d'attribut pour cette AVP a le format suivant :

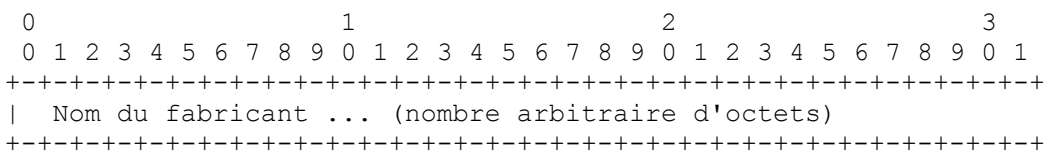


Identifiant de routeur est un entier non signé de 4 octets. Sa valeur est unique pour un certain LCCE, selon le paragraphe 8.1 de la [RFC2072]. Les AVP Nom d'hôte et/ou Identifiant de routeur DOIVENT être utilisées pour identifier un LCCE comme décrit au paragraphe 3.3.

Les mises en œuvre NE DOIVENT PAS supposer que l'identifiant de routeur est une adresse IP valide. L'identifiant de routeur pour L2TP sur IPv6 peut être obtenu d'une adresse IPv4 (si disponible) ou via des moyens non spécifiés spécifiques de la mise en œuvre.

Cette AVP NE DOIT PAS être cachée (le bit H DOIT être 0). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur de cette AVP est 10.

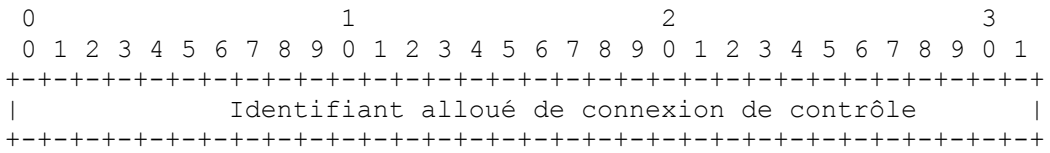
Nom du fabricant (SCCRQ, SCCRP)
L'AVP Nom du fabricant, type d'attribut 8, contient une chaîne spécifique du fabricant (éventuellement lisible par l'homme) qui décrit le type de LAC ou LNS utilisé. Le champ Valeur d'attribut pour cette AVP a le format suivant :



Nom du fabricant est le nombre d'octets indiqué qui représente la chaîne du fabricant. Le texte lisible par l'homme pour cette AVP DOIT être fourni dans le jeu de caractères US-ASCII [RFC1958], [RFC2277].

Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 0, mais PEUT varier (voir au paragraphe 5.2). Le champ Longueur (avant de cacher) de cette AVP est 6 plus la longueur du Nom du fabricant.

Identifiant alloué de connexion de contrôle (SCCRQ, SCCRP, StopCCN)
L'AVP Identifiant alloué de connexion de contrôle, type d'attribut 61, contient l'identifiant qui est alloué à cette connexion de contrôle par l'envoyeur. Le champ Valeur d'attribut pour cette AVP a le format suivant :



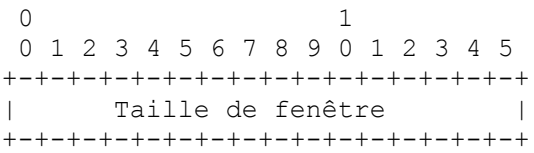
L'identifiant alloué de connexion de contrôle est un entier non signé de 4 octets non à zéro.

L'AVP Identifiant alloué de connexion de contrôle établit l'identifiant utilisé pour multiplexer et démultiplexer plusieurs connexions de contrôle entre une paire de LCCE. Une fois que l'AVP Identifiant alloué de connexion de contrôle a été reçue par un LCCE, l'identifiant de connexion de contrôle spécifié dans l'AVP DOIT être inclus dans le champ Identifiant de connexion de contrôle de tout paquet de commande envoyé à l'homologue pour la durée de vie de la connexion de contrôle. Avant que l'AVP Identifiant alloué de connexion de contrôle ne soit reçue d'un homologue, tous les messages de contrôle DOIVENT être envoyés à cet homologue avec une valeur d'identifiant de connexion de contrôle de 0 dans l'en-tête. Parce que une valeur d'identifiant de connexion de contrôle de 0 est utilisée de cette façon particulière, la valeur zéro NE DOIT PAS être envoyée comme valeur d'identifiant alloué de connexion de contrôle.

Dans certaines circonstances, un LCCE peut devoir envoyer un StopCCN à un homologue sans avoir encore reçu une AVP Identifiant alloué de connexion de contrôle de la part de l'homologue (c'est-à-dire, SCCRQ envoyé, pas encore de SCCRP reçu). Dans ce cas, l'AVP Identifiant alloué de connexion de contrôle qui avait été envoyée précédemment à l'homologue (c'est-à-dire, dans le SCCRQ) DOIT être envoyée comme AVP Identifiant alloué de connexion de contrôle dans le StopCCN. Cette politique permet à l'homologue d'essayer d'identifier la connexion de contrôle appropriée via une recherche inverse.

Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur de cette AVP (avant d'être cachée) est 10.

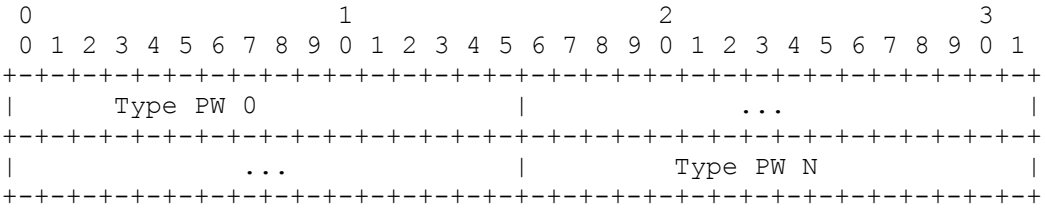
Taille de la fenêtre de réception (SCCRQ, SCCRP)
L'AVP Taille de la fenêtre de réception, type d'attribut 10, spécifie la taille de la fenêtre de réception offerte à l'homologue distant. Le champ Valeur d'attribut pour cette AVP a le format suivant :



Taille de fenêtre est un entier non signé de deux octets. S'il est absent, l'homologue doit supposer une taille de fenêtre de 4 pour sa fenêtre de transmission. L'homologue distant peut envoyer le nombre de messages de contrôle spécifié avant qu'il doive attendre un accusé de réception. Voir au paragraphe 4.2 plus d'informations sur la livraison fiable de message de contrôle.

Cette AVP NE DOIT PAS être cachée (le bit H DOIT être 0). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur de cette AVP est 8.

Liste de capacités pseudo filaires (SCCRQ, SCCRP)
L'AVP Liste de capacités pseudo filaires, type d'attribut 62, indique les types de charge utile de couche 2 que peut accepter l'envoyeur. Le type de charge utile spécifique d'une certaine session est identifié par l'AVP Type pseudo filaire. Le champ Valeur d'attribut pour cette AVP a le format suivant :

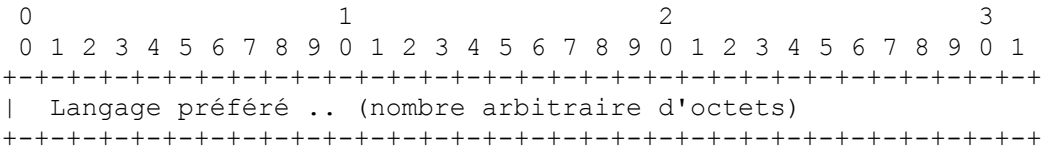


Les types PW définis qui peuvent apparaître dans cette liste sont gérés par l'IANA et vont apparaître dans les documents associés spécifiques de pseudo filaire pour chaque type PW.

Si un envoyeur inclut un certain type PW dans l'AVP Liste de capacités pseudo filaires, l'envoyeur assume la pleine responsabilité de la prise en charge de cette charge utile particulière, comme toutes AVP spécifique de charge utile, sous couche spécifique de couche 2, ou messages de contrôle qui peuvent être définis dans les documents d'accompagnement appropriés.

Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 8 octets avec un type pseudo filaire spécifié, plus 2 octets pour chaque type PW additionnel.

Langage préféré (SCCRQ, SCCRP)
L'AVP Langage préféré, type d'attribut 72, donne une méthode pour qu'un LCCE indique à son homologue le langage dans lequel DEVRAIENT être composés les messages lisibles par l'homme qu'il envoie. Cette AVP contient une seule étiquette de langage ou gamme de langages [RFC3066]. Le champ Valeur d'attribut pour cette AVP a le format suivant :

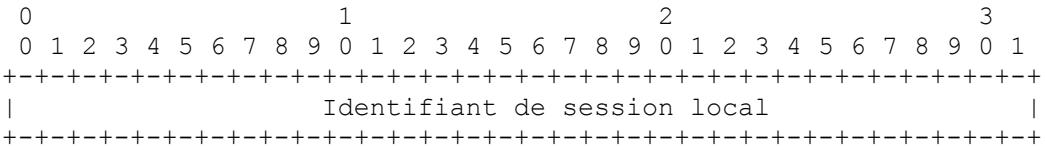


Langage préféré est le nombre d'octets indiqué qui représente l'étiquette de langage ou gamme de langage, codée dans le jeu de caractères US-ASCII. Il n'est pas exigé que soit envoyée une AVP Langage préféré. Si (1) un LCCE ne signifie pas de préférence de langage par l'inclusion de cette AVP dans le SCCRQ ou SCCRP, (2) l'AVP Langage préféré n'est pas reconnue, ou (3) le langage demandé n'est pas pris en charge par le LCCE homologue, le langage par défaut de la [RFC2277] DOIT être utilisé pour toutes les chaînes internationalisées envoyées par l'homologue.

Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 0, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 6 plus la longueur du langage préféré.

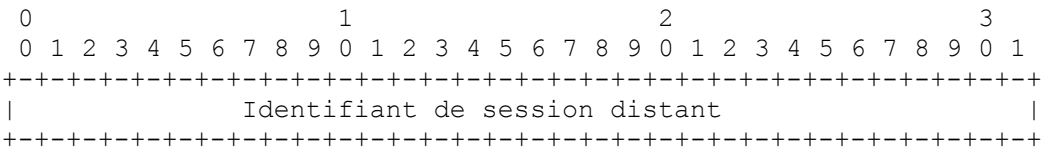
5.4.4 AVP de gestion de session

Identifiant de session local (ICRQ, ICRP, ICCN, OCRQ, OCRP, OCCN, CDN, WEN, SLI)
L'AVP Identifiant de session local (analogue à l'identifiant de session allouée dans L2TPv2), type d'attribut 63, contient l'identifiant qui est alloué à cette session par l'envoyeur. Le champ Valeur d'attribut pour cette AVP a le format suivant :



L'identifiant de session local est un entier non signé de 4 octets différent de zéro. L'AVP Identifiant de session local établit les deux identifiants utilisés pour multiplexer et démultiplexer les sessions entre deux LCCE. Chaque LCCE choisit librement la valeur qu'il désire, et l'envoie au LCCE distant en utilisant cette AVP. Le LCCE distant DOIT alors envoyer tous les paquets de données associés à cette session en utilisant cette valeur. De plus, pour tous les messages de contrôle en mode session envoyés après que cette AVP est reçue (par exemple, ICRP, ICCN, CDN, SLI, etc.) le LCCE distant DOIT faire écho à cette valeur dans l'AVP Identifiant de session distant. Noter qu'une valeur d'identifient de session est unidirectionnelle. Parce que chaque LCCE choisit son identifiant de session indépendamment de son LCCE homologue, la valeur n'a pas besoin de correspondre dans chaque direction pour une session donnée. Voir au paragraphe 4.1 des informations supplémentaires sur l'identifiant de session. Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 10.

Identifiant de session distante (ICRQ, ICRP, ICCN, OCRQ, OCRP, OCCN, CDN, WEN, SLI)
L'AVP Identifiant de session distant, type d'attribut 64, contient l'identifiant qui a été alloué à cette session par l'homologue. Le champ Valeur d'attribut pour cette AVP a le format suivant :

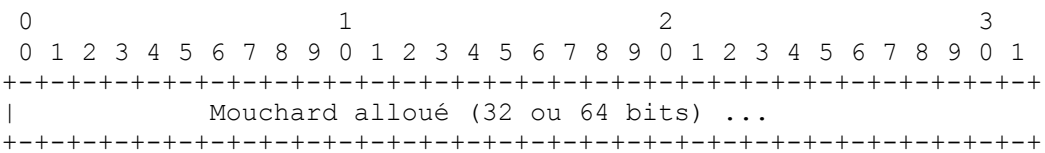


L'identifiant de session distant est un entier non signé de 4 octets différent de zéro. L'AVP Identifiant de session distant DOIT être présente dans tous les messages de contrôle de niveau session. La valeur de l'AVP fait écho à l'identifiant de session annoncé par l'homologue via l'AVP Identifiant de session local. C'est la même valeur qui va être utilisée dans tous les messages de données transmis par ce côté de la session. Dans la plupart des cas, cet identifiant est suffisant pour que l'homologue trouve le contexte de niveau session pour ce message de contrôle.

Lorsque un message de contrôle de niveau session doit être envoyé à l'homologue avant que l'AVP Identifiant de session local ait été reçu, la valeur de l'AVP Identifiant de session distant DOIT être réglée à zéro. De plus, l'AVP Identifiant de session local (envoyée dans un message de contrôle précédent pour cette session) DOIT être incluse dans le message de contrôle. L'homologue doit alors utiliser l'AVP Identifiant de session local pour effectuer la recherche inverse pour trouver son contexte de session. Les messages de contrôle de niveau session définis dans le présent document qui peuvent être soumis à une recherche inverse par un homologue receveur incluent CDN, WEN, et SLI.

Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 10.

Mouchard alloué (ICRQ, ICRP, OCRQ, OCRP)
L'AVP Mouchard alloué, type d'attribut 65, contient la valeur du mouchard qui est allouée à cette session par l'envoyeur. Le champ Valeur d'attribut pour cette AVP a le format suivant :

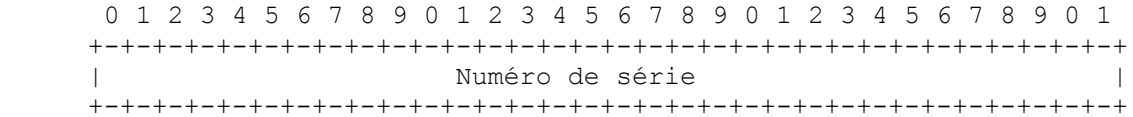


Le mouchard alloué est une valeur aléatoire de 4 ou 8 octets. L'AVP Mouchard alloué contient la valeur utilisée pour vérifier l'association d'un message de données reçu par rapport à la session identifiée par l'identifiant de session. Tous les messages de données envoyés à un homologue DOIVENT utiliser le mouchard alloué envoyé par l'homologue dans cette AVP. La longueur de la valeur (0, 32, ou 64 bits) est obtenue par la longueur de l'AVP.

Une AVP Mouchard alloué manquante ou une valeur de mouchard alloué de longueur zéro indique que le champ Cookie ne devrait être présent dans aucun paquet de données envoyé au LCCE qui envoie cette AVP. Voir au paragraphe 4.1 des informations supplémentaires sur le mouchard alloué.

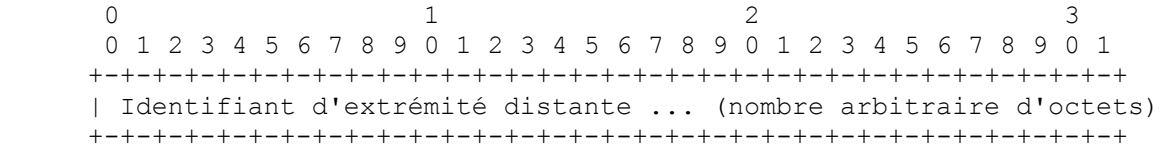
Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP peut être 6, 10, ou 14 octets.

Numéro de série (ICRQ, OCRQ)
L'AVP Numéro de série, type d'attribut 15, contient un identifiant alloué par le LAC ou le LNS à cette session. Le champ Valeur d'attribut pour cette AVP a le format suivant :



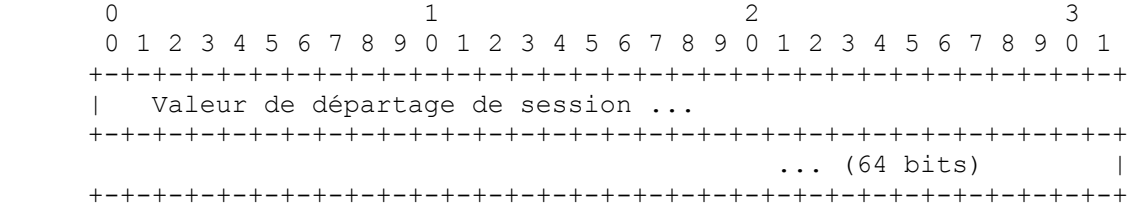
Le numéro de série est une valeur de 32 bits. Le numéro de série est destiné à être une référence facile pour les administrateurs des deux côtés d'une connexion de contrôle à utiliser pour enquêter sur des problèmes d'échec de session. Les numéros de série devraient être établis à des valeurs de croissance progressive, qui devraient être uniques pendant une durée significative entre tous les LNS et LAC interconnectés. Noter que dans la RFC 2661, cette valeur était appelée "AVP Numéro de série d'appel". Elle sert au même objet et a la même valeur d'attribut et composition. Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 0, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 10.

Identifiant d'extrémité distante (ICRQ, OCRQ)
L'AVP Identifiant d'extrémité distante, type d'attribut 66, contient un identifiant utilisé pour lier les sessions L2TP à un certain circuit, interface, ou instance de pontage. Elle peut aussi être utilisée pour détecter des liens de niveau session. Le champ Valeur d'attribut pour cette AVP a le format suivant :



Le champ Identifiant d'extrémité distante est de longueur variable et sa valeur est unique pour chaque LCCE homologue, comme décrit au paragraphe 3.3. Un nœud de niveau session est détecté si un LCCE reçoit un ICRQ ou OCRQ avec une AVP Identifiant d'extrémité dont la valeur correspond à celle qui a juste été envoyée au même homologue dans un ICRQ ou OCRQ sortant. Si les deux valeurs correspondent, un LCCE reconnaît qu'un nœud existe (c'est-à-dire, que les deux LCCE tentent d'établir des sessions pour le même circuit). Le nœud est rompu par l'AVP Départage de session. Par défaut, une application d'interconnexion de LAC à LAC (voir le point (b) de la Section 2) de L2TP sur un réseau IP DOIT utiliser l'AVP Identifiant de routeur et l'AVP Identifiant d'extrémité distante pour associer un circuit à une session L2TP. D'autres AVP PEUVENT être utilisées pour l'identification de LCCE ou de circuit comme spécifié dans les documents d'accompagnement. Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 6 plus la longueur de la valeur de l'identifiant d'extrémité distante.

Départage de session (ICRQ, OCRQ)
L'AVP Départage de session, type d'attribut 5, est utilisée pour casser les nœuds lorsque deux homologues tentent concurremment d'établir une session sur le même circuit. Le champ Valeur d'attribut pour cette AVP a le format suivant :



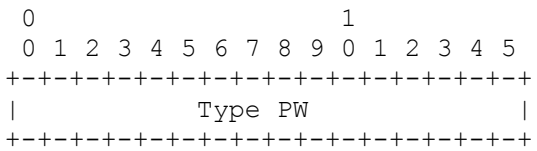
La valeur de départage de session est une valeur aléatoire de 8 octets qui est utilisée pour choisir une session lorsque deux LCCE demandent concurremment une session pour le même circuit. Un nœud est détecté en examinant l'identité de l'homologue (décrite au paragraphe 3.3) plus la valeur partagée par session communiquée via l'AVP Identité d'extrémité. Dans le cas d'un nœud, le receveur d'un ICRQ ou OCRQ doit comparer la valeur de départage reçue à celle qu'il a envoyé

antérieurement. Le LCCE qui a la valeur la plus faible "gagne" et DOIT envoyer un CDN avec le code de résultat réglé à 13 (comme défini au paragraphe 5.4.2) en réponse au ICRQ ou OCRQ perdant. Dans le cas où un nœud est détecté, les départages sont envoyés par les deux côtés, et les valeurs de départage sont égales, les deux côtés DOIVENT supprimer leurs sessions et recommencer la négociation de session avec de nouvelles valeurs aléatoires de départage.

Si un nœud est détecté mais si un seul côté envoie une AVP Départage de session, l'initiateur de la session qui a inclus l'AVP Départage de session "gagne". Si aucun ne produit de départage, tous deux DOIVENT supprimer la session.

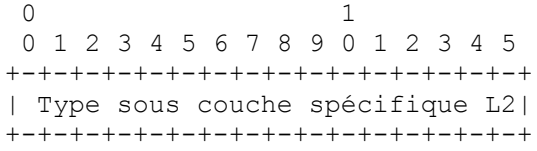
Cette AVP NE DOIT PAS être cachée (le bit H DOIT être 0). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur de cette AVP est 14.

Type pseudo filaire (ICRQ, OCRQ)
L'AVP Type pseudo filaire (Type PW), type d'attribut 68, indique le type de charge utile de couche 2 des paquets qui vont être tunnelés en utilisant cette session L2TP. Le champ Valeur d'attribut pour cette AVP a le format suivant :



Un homologue NE DOIT PAS demander un appel entrant ou sortant avec une AVP Type PW spécifiant une valeur non annoncée dans l'AVP Liste de capacités pseudo filaires qu'il a reçue durant l'établissement de la connexion de contrôle. Les tentatives de le faire DOIVENT résulter en un rejet d'appel via un CDN avec le code de résultat réglé à 14 (voir au paragraphe 5.4.2). Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 8.

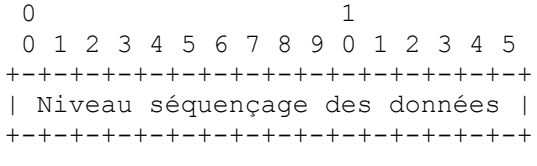
Sous couche spécifique de couche 2 (ICRQ, ICRP, ICCN, OCRQ, OCRP, OCCN)
L'AVP Sous couche spécifique de couche 2, type d'attribut 69, indique la présence et le format de la sous couche spécifique de couche 2 que l'envoyeur de cette AVP exige sur tous les paquets de données entrants pour cette session L2TP.



Le type de sous couche spécifique de couche 2 est un entier non signé de deux octets avec les valeurs suivantes définies dans le présent document :
0 – Aucune sous couche spécifique de couche 2 n'est présente.
1 – On utilise la sous couche spécifique de couche 2 par défaut (définie au paragraphe 4.6).

Si cette AVP est reçue et a une valeur autre que zéro, le LCCE receveur DOIT inclure la sous couche spécifique de couche 2 identifiée dans ses messages de données sortants. Si l'AVP n'est pas reçue, on suppose qu'aucune sous couche n'est présente. Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 8.

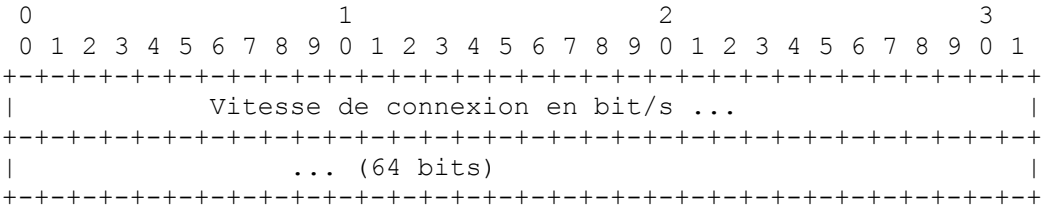
Séquençage de données (ICRQ, ICRP, ICCN, OCRQ, OCRP, OCCN)
L'AVP Séquençage de données, type d'attribut 70, indique que l'envoyeur exige que tout ou partie des paquets de données qu'il reçoit soient marqués par un numéro de séquence. Le champ Valeur d'attribut pour cette AVP a le format suivant :



Le niveau de séquençage des données est un entier non signé de deux octets indiquant le degré auquel l'envoyeur de cette AVP souhaite que le trafic de données entrant soit marqué avec des numéros de séquence. Les niveaux définis de séquençage des sonnées sont comme suit :
0 – Aucun paquet de données entrant n'exige le séquençage.
1 – Seuls les paquets de données non IP exigent le séquençage.
2 – Tous les paquets de données entrants exigent le séquençage.

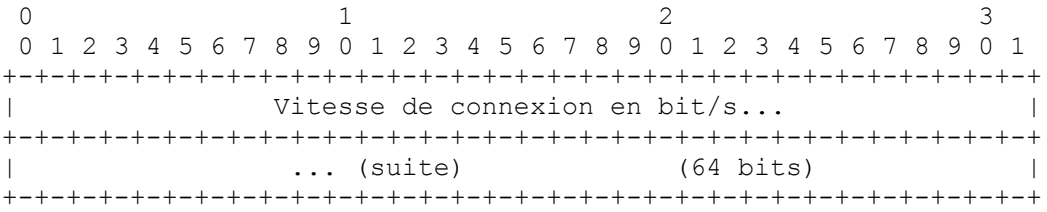
Si le niveau 0 de séquençage des données est spécifié, il n'est pas nécessaire d'envoyer des paquets avec des numéros de séquence. Si des numéros de séquence sont envoyés, ils seront ignorés à réception. Si on ne reçoit pas d'AVP Séquençage des données, on suppose le niveau de séquençage de 0. Si un niveau de séquençage des données de 1 est spécifié, seul le trafic non IP porté dans la trame L2 tunnelée devrait se voir appliquer des numéros de séquence. Ici, trafic non IP se réfère à tous les paquets qui ne peuvent pas être classéd comme paquetd IP dans leurs tramages L2 respectifs (par exemple, un paquet de contrôle PPP ou une trame NETBIOS encapsulée par relais de trame avant d'être tunnelée). Tout le trafic qui peut être classé comme IP DOIT être envoyé sans séquençage (c'est-à-dire, le bit S dans la sous couche spécifique de L2 est réglé à zéro). Si un paquet ne peut pas être classé du tout (par exemple, parce que il a été compressé ou chiffré à la couche 2) ou si une mise en œuvre est incapable d'effectuer une telle classification au sein des trames L2, tous les paquets DOIVENT être munis d'un numéro de séquence (ce qui revient essentiellement au niveau 2 de séquençage des données). Si un niveau de séquençage des données de 2 est spécifié, tout le trafic DOIT être muni de numéros de séquence. Le séquençage des données ne peut être exigé que lorsque une sous couche spécifique de couche 2 est présente qui peut fournir des numéros de séquence. Si le séquençage est exigé sans demander une AVP Sous couche spécifique de couche 2, la session DOIT être déconnectée avec un code de résultat de 15 (voir au paragraphe 5.4.2).
Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 8.

Vitesse de connexion en émission (ICRQ, ICRP, ICCN, OCRQ, OCRP, OCCN)
L'AVP Vitesse de connexion en émission en bits par seconde, type d'attribut 74, contient la vitesse de la facilité choisie pour la tentative de connexion. Le champ Valeur d'attribut pour cette AVP a le format suivant :



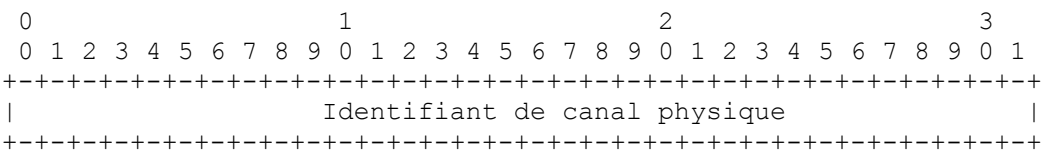
Vitesse de connexion en émission en bits par seconde est une valeur de 8 octets qui indique en bits par seconde la vitesse de transmission. Une valeur de zéro indique que la vitesse est indéterminée ou qu'il n'y a pas de liaison physique point à point. Lorsque l'AVP facultative Vitesse de connexion en réception est présente, la valeur dans cette AVP représente la vitesse de connexion en émission du point de vue du LAC (c'est-à-dire, les données qui s'écoulent du LAC vers le système distant). Lorsque l'AVP facultative Vitesse de connexion en réception N'EST PAS présente, la vitesse de connexion entre le système distant et le LAC est supposée être symétrique et est représentée par la seule valeur de cette AVP. Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 0, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 14.

Vitesse de connexion en réception (ICRQ, ICRP, ICCN, OCRQ, OCRP, OCCN)
L'AVP Vitesse de connexion en réception , type d'attribut 75, représente la vitesse de la connexion du point de vue du LAC (c'est-à-dire, des données qui s'écoulent du système distant au LAC). Le champ Valeur d'attribut pour cette AVP a le format suivant :



Vitesse de connexion en bit/s est une valeur de 8 octets qui indique la vitesse en bits par seconde. Une valeur de zéro indique que la vitesse est indéterminée ou qu'il n'y a pas de liaison physique point à point. La présence de cette AVP implique que la vitesse de connexion peut être asymétrique par rapport à la vitesse de connexion en émission donnée dans l'AVP Vitesse de connexion en émission. Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 0, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 14.

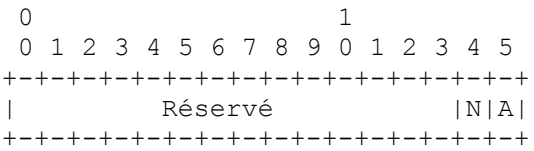
Identifiant de canal physique (ICRQ, ICRP, OCRP)
L'AVP Identifiant de canal physique, type d'attribut 25, contient le numéro de canal physique spécifique du fabricant qui est utilisé pour un appel. Le champ Valeur d'attribut pour cette AVP a le format suivant :



Identifiant de canal physique est une valeur de quatre octets destinée à n'être utilisée que pour des besoins de tenue de journal d'événements. Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 0, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 10.

5.4.5 AVP d'état de circuit

État de circuit (ICRQ, ICRP, ICCN, OCRQ, OCRP, OCCN, SLI)
L'AVP État de circuit, type d'attribut 71, indique l'état initial ou le changement d'état du circuit auquel la session est liée. Le champ Valeur d'attribut pour cette AVP a le format suivant :



Le bit A (Actif) indique si le circuit est actif/prêt (1) ou inactif/pas prêt (0).
Le bit N (Nouveau) indique si l'indication d'état de circuit est pour un nouveau circuit (1) ou un circuit existant (0). Les liaisons qui ont la disposition d'un mécanisme similaire (par exemple, relais de trame) DOIVENT transposer le réglage de ce bit en la signalisation associée pour cette liaison. Autrement, le bit Nouveau DEVRAIT être établi la première fois que la session L2TP est établie après le provisionnement.
Les bits restants sont réservés pour une utilisation future. Les bits réservés DOIVENT être à 0 à l'envoi et ignorés à réception.

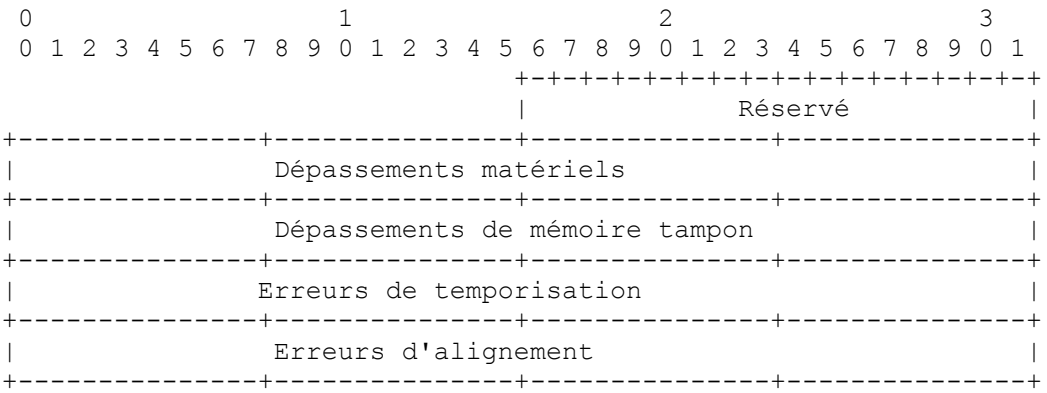
L'AVP État de circuit est utilisée pour annoncer si un circuit ou interface lié à une session L2TP est actif et prêt à envoyer et/ou recevoir du trafic. Les différents types de circuits ont des noms différents pour les types d'état. Par exemple, les stations primaires et secondaires HDLC se réfèrent à un circuit comme étant "Prêt à recevoir" ou "Pas prêt à recevoir", tandis que le relais de trame se réfère à un circuit comme "Actif" ou "Inactif". Cette AVP adopte cette dernière terminologie, bien que le concept reste le même sans considération du type PW pour la session L2TP.

Dans le cas le plus simple, le circuit auquel se réfère cette AVP est une seule interface physique, accès, ou circuit, selon l'application et l'établissement de la session. L'indication d'état dans cette AVP peut alors être utilisée pour assurer l'interfonctionnement de simple interface de gestion locale interne pour divers types de circuits. Pour des interfaces virtuelles ou multipoint, l'AVP État de circuit est aussi utilisée, mais dans ce cas, elle se réfère à l'état d'une structure interne ou un ensemble logique de circuits. Chaque document d'accompagnement spécifique de PW DOIT spécifier précisément comment cette AVP est traduite pour chaque type de circuit.

Si cette AVP est reçue avec une notification Non Actif pour une certaine session L2TP, tout le trafic de données pour cette session DOIT cesser (ou ne pas commencer) dans la direction de l'envoyeur de l'AVP État de circuit jusqu'à ce que le circuit soit annoncé comme Actif.

L'état de circuit DOIT être annoncé par cette AVP dans les messages ICRQ, ICRP, OCRQ, et OCRP. Souvent, le type de circuit sera marqué Actif à l'initialisation, mais PEUT ensuite être annoncé comme Inactif. Cela indique qu'une session L2TP va être créée, mais que l'interface ou circuit n'est pas encore prêt à passer du trafic. Les messages de contrôle ICCN, OCCN, et SLI PEUVENT tous contenir cette AVP pour mettre à jour l'état du circuit après la demande d'établissement de la session L2TP. Si des informations d'état de circuit supplémentaires sont nécessaires pour un certain type de PW, toute nouvelle AVP spécifique du PW DOIT être définie dans un document séparé. Cette AVP est seulement pour les informations générales d'état de circuit applicables à tous les types de circuit/interface. Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 1, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 8.

Erreurs de circuit (WEN)
L'AVP Erreur de circuit, type d'attribut 34, transporte les informations d'erreur de circuit à l'homologue. Le champ Valeur d'attribut pour cette AVP a le format suivant :



Les champs suivants sont définis :
Réservé : 2 octets de données réservés sont présents (fournissant un alignement de longueur de mot au sein de l'AVP des valeurs suivantes). Les données réservées DOIVENT être à zéro à l'envoi et ignorées à réception.
Dépassements matériels : Nombre de dépassement de mémoire tampon de réception depuis l'établissement de l'appel.
Dépassements de mémoire tampon : Nombre de dépassements de mémoire tampon détectés depuis l'établissement de l'appel.
Erreurs de temporisation : Nombre de fins de temporisation depuis l'établissement de l'appel.
Erreurs d'alignement : Nombre d'erreurs d'alignement depuis que l'appel a été établi.

Cette AVP PEUT être cachée (le bit H PEUT être 0 ou 1). Le bit M pour cette AVP DEVRAIT être réglé à 0, mais PEUT varier (voir au paragraphe 5.2). La longueur (avant d'être cachée) de cette AVP est 32.

6. Spécification du protocole de connexion de contrôle

Les messages de contrôle suivants sont utilisés pour établir, maintenir, et supprimer les connexions de contrôle L2TP. Tous les paquets de données sont envoyés dans l'ordre des octets du réseau (octets de poids fort en premier). Tous les champs "réservé" ou "vide" DOIVENT être envoyés avec des valeurs de 0 pour permettre l'extensibilité du protocole.

Les échanges dans lesquels ces messages sont impliqués sont mentionnés au paragraphe 3.3.

6.1 Demande de début de connexion de contrôle (SCCRQ, *Start-Control-Connection-Request*)

Start-Control-Connection-Request (SCCRQ) est un message de contrôle utilisé pour initier une connexion de contrôle entre deux LCCE. Il est envoyé par le LAC ou par le LNS pour commencer le processus d'établissement de la connexion de contrôle.

Les AVP suivantes DOIVENT être présentes dans la SCCRQ :
Type de message
Nom d'hôte
Identifiant de routeur
Identifiant alloué de connexion de contrôle
Liste des capacités pseudo filaires

Les AVP suivantes PEUVENT être présentes dans la SCCRQ :
Vecteur aléatoire
Nom occasionnel d'authentification de message de contrôle
Résumé de message
Départage de connexion de contrôle
Nom du fabricant
Taille de la fenêtre de réception
Langage préféré

6.2 Réponse de début de connexion de contrôle (SCCRP, *Start-Control-Connection-Reply*)

Start-Control-Connection-Reply (SCCRP) est le message de contrôle envoyé en réponse à un message SCCRQ reçu. La SCCRP est utilisée pour indiquer que la SCCRQ a été acceptée et que l'établissement de la connexion de contrôle devrait continuer.

Les AVP suivantes DOIVENT être présentes dans la SCCRP :
Type de message
Nom d'hôte
Identifiant de routeur
Identifiant alloué de connexion de contrôle
Liste des capacités pseudo filaires

Les AVP suivantes PEUVENT être présentes dans la SCCRP :
Vecteur aléatoire
Nom occasionnel d'authentification de message de contrôle
Résumé de message
Nom du fabricant
Taille de la fenêtre de réception
Langage préféré

6.3 Début de connexion de contrôle connectée (SCCCN, *Start-Control-Connection-Connected*)

Start-Control-Connection-Connected (SCCCN) est le message de contrôle envoyé en réponse à une SCCRP. Le SCCCN achève le processus d'établissement de la connexion de contrôle.

L'AVP suivante DOIT être présente dans le SCCCN :
Type de message

Les AVP suivantes PEUVENT être présentes dans le SCCCN :
Vecteur aléatoire
Résumé de message

6.4 Notification d'arrêt de connexion de contrôle (StopCCN, *Stop-Control-Connection-Notification*)

Stop-Control-Connection-Notification (StopCCN) est le message de contrôle envoyé par un LCCE pour informer son homologue de la fermeture de la connexion de contrôle et que la connexion de contrôle devrait être close. De plus, toutes les sessions actives sont implicitement supprimées (sans envoyer de message de contrôle de session explicite). La raison de la production de cette demande est indiquée dans l'AVP de code de résultat. Il n'y a pas de réponse explicite à ce message, seulement l'accusé de réception implicite qui est reçu par la couche de livraison fiable de message de contrôle.

Les AVP suivantes DOIVENT être présentes dans la StopCCN :
Type de message
Code de résultat

Les AVP suivantes PEUVENT être présentes dans la StopCCN :
Vecteur aléatoire
Résumé de message
Identifiant alloué de connexion de contrôle

Noter que l'Identifiant alloué de connexion de contrôle DOIT être présent si la StopCCN est envoyée après l'envoi d'un message SCCRQ ou SCCRP.

6.5 Hello (HELLO)

Le message Hello (HELLO) est un message de contrôle L2TP envoyé par l'un ou l'autre homologue d'une connexion de contrôle. Ce message de contrôle est utilisé comme moyen de "garder en vie" la connexion de contrôle. Voir au paragraphe 4.2 la description du mécanisme de maintien en vie.

Les messages HELLO sont globaux pour la connexion de contrôle. L'identifiant de session dans un message HELLO DOIT être 0.

L'AVP suivante DOIT être présente dans le HELLO :
Type de message

Les AVP suivantes PEUVENT être présentes dans le HELLO :
Vecteur aléatoire
Résumé de message

6.6 Demande d'appel entrant (ICRQ, *Incoming-Call-Request*)

Incoming-Call-Request (ICRQ) est le message de contrôle envoyé par un LCCE à un homologue lorsque un appel entrant est détecté (bien que l'ICRQ puisse aussi être envoyée par suite d'un événement local). C'est le premier d'un échange de trois messages utilisé pour établir une session via une connexion de contrôle L2TP.

ICRQ est utilisée pour indiquer qu'une session doit être établie entre un LCCE et un homologue. L'envoyeur d'une ICRQ fournit à l'homologue des informations de paramètres pour la session. Cependant, l'envoyeur ne fait pas de demande sur la façon dont la session se termine chez l'homologue (c'est-à-dire, si le trafic L2 est traité en local, transmis, etc.).

Les AVP suivantes DOIVENT être présentes dans l'ICRQ :
Type de message
Identifiant de session locale
Identifiant de session distante
Numéro de série
Type pseudo filaire
Identifiant d'extrémité distante
État de circuit

Les AVP suivantes PEUVENT être présentes dans l'ICRQ :
Vecteur aléatoire
Résumé de message
Mouchard alloué
Départage de session
Sous couche spécifique de couche 2
Séquençage de données
Vitesse de connexion en émission
Vitesse de connexion en réception
Identifiant de canal physique

6.7 Réponse d'appel entrant (ICRP, *Incoming-Call-Reply*)

Incoming-Call-Reply (ICRP) est le message de contrôle envoyé par un LCCE en réponse à une ICRQ reçue. C'est le second dans l'échange de trois messages utilisé pour établir une session au sein d'une connexion de contrôle L2TP.

ICRP est utilisé pour indiquer que l'ICRQ a réussi et que l'homologue devrait établir (c'est-à-dire, répondre à) l'appel entrant si il ne l'a déjà fait. Il permet aussi à l'envoyeur d'indiquer des paramètres spécifiques sur la session L2TP.

Les AVP suivantes DOIVENT être présentes dans la ICRP :
Type de message
Identifiant de session locale
Identifiant de session distante
État de circuit

Les AVP suivantes PEUVENT être présentes dans la ICRP :
Vecteur aléatoire
Résumé de message
Mouchard alloué
Sous couche spécifique de couche 2
Séquençage de données
Vitesse de connexion en émission
Vitesse de connexion en réception
Identifiant de canal physique

6.8 Appel entrant connecté (ICCN, *Incoming-Call-Connected*)

Incoming-Call-Connected (ICCN) est le message de contrôle envoyé par le LCCE qui a envoyé à l'origine une ICRQ à réception d'une ICRP de son homologue. C'est le message final dans l'échange de trois messages utilisé pour établir les sessions L2TP.

ICCN est utilisé pour indiquer que l'ICRP a été acceptée, que l'appel a été établi, et que la session L2TP devrait passer à l'état

Établi. Il permet aussi à l'envoyeur d'indiquer des paramètres spécifiques sur l'appel établi (des paramètres qui peuvent n'avoir pas été disponibles au moment de la production de l'ICRQ).

Les AVP suivantes DOIVENT être présentes dans l'ICCN :
Type de message
Identifiant de session locale
Identifiant de session distante

Les AVP suivantes PEUVENT être présentes dans l'ICCN :
Vecteur aléatoire
Résumé de message
Sous couche spécifique de couche 2
Séquençage de données
Vitesse de connexion en émission
Vitesse de connexion en réception
État de circuit

6.9 Demande d'appel sortante (OCRQ, *Outgoing-Call-Request*)

Outgoing-Call-Request (OCRQ) est le message de contrôle envoyé par un LCCE à un LAC pour indiquer qu'un appel sortant pour le LAC va être établi sur la base des informations de destination spécifiques envoyées dans ce message. C'est le premier d'un échange de trois messages utilisé pour établir une session et passer un appel au nom du LCCE initiateur.

Noter qu'un appel peut être toute connexion L2 exigeant que des informations de destination bien connues soient envoyées d'un LCCE à un LAC. Cet appel pourrait être une connexion numérotée sur le RTPC, une connexion SVC, l'adresse IP d'un autre LCCE, ou toute autre destination dictée par l'envoyeur de ce message.

Les AVP suivantes DOIVENT être présentes dans la OCRQ :
Type de message
Identifiant de session locale
Identifiant de session distante
Numéro de série
Type pseudo filaire
Identifiant d'extrémité distante
État de circuit

Les AVP suivantes PEUVENT être présentes dans la OCRQ :
Vecteur aléatoire
Résumé de message
Mouchard alloué
Vitesse de connexion en émission
Vitesse de connexion en réception
Départage de session
Sous couche spécifique de couche 2
Séquençage de données

6.10 Réponse d'appel sortant (OCRP, *Outgoing-Call-Reply*)

Outgoing-Call-Reply (OCRP) est le message de contrôle envoyé par un LAC à un LCCE en réponse à la réception d'une OCRQ. C'est le second d'un échange de trois messages utilisé pour établir une session au sein d'une connexion de contrôle L2TP.

OCRP est utilisé pour indiquer que le LAC a été capable de tenter l'appel sortant. Le message retourne tous les paramètres pertinents concernant la tentative d'appel. Les données NE DOIVENT PAS être transmises avant que l'OCCN soit reçue, qui indiquera que l'appel a été passé.

Les AVP suivantes DOIVENT être présentes dans l'OCRP :
Type de message
Identifiant de session locale
Identifiant de session distante
État de circuit

Les AVP suivantes PEUVENT être présentes dans l'OCRP :
Vecteur aléatoire
Résumé de message
Mouchard alloué
Sous couche spécifique de couche 2
Vitesse de connexion en émission
Vitesse de connexion en réception
Séquençage de données
Identifiant de canal physique

6.11 Appel sortant connecté (OCCN, *Outgoing-Call-Connected*)

Outgoing-Call-Connected (OCCN) est le message de contrôle envoyé par un LAC à un autre LCCE après la OCRP et après que l'appel sortant a été achevé. C'est le message final de l'échange de trois messages utilisé pour établir une session.

OCCN est utilisé pour indiquer que le résultat de l'appel sortant demandé est réussi. Il donne aussi des information au LCCE qui a demandé l'appel sur les paramètres particuliers obtenus après l'établissement de l'appel.

Les AVP suivantes DOIVENT être présentes dans OCCN :
Type de message
Identifiant de session locale
Identifiant de session distante

Les AVP suivantes PEUVENT être présentes dans OCCN :
Vecteur aléatoire
Résumé de message
Sous couche spécifique de couche 2
Vitesse de connexion en émission
Vitesse de connexion en réception
Séquençage de données
État de circuit

6.12 Notification de déconnexion d'appel (CDN, *Call-Disconnect-Notify*)

Call-Disconnect-Notify (CDN) est un message de contrôle envoyé par un LCCE pour demander la déconnexion d'une session spécifique. Son objet est d'informer l'homologue de la déconnexion et de la raison de la déconnexion. L'homologue DOIT nettoyer toutes ses ressources, et ne renvoie aucune indication de succès ou d'échec pour un tel nettoyage.

Les AVP suivantes DOIVENT être présentes dans la CDN :
Type de message
Code de résultat
Identifiant de session locale
Identifiant de session distante

Les AVP suivantes PEUVENT être présentes dans la CDN :
Vecteur aléatoire
Résumé de message

6.13 Notification d'erreur de WAN (WEN, *WAN-Error-Notify*)

WAN-Error-Notify (WEN) est un message de contrôle envoyé d'un LAC à un LNS pour indiquer des conditions d'erreur de WAN. Les compteurs dans ce message sont cumulatifs. Ce message ne devrait être envoyé que lorsque survient une erreur, et pas plus d'une fois toutes les 60 secondes. Les compteurs sont remis à zéro lorsque un nouvel appel est établi.
Les AVP suivantes DOIVENT être présentes dans WEN :
Type de message
Identifiant de session locale
Identifiant de session distante
Erreur de circuit

Les AVP suivantes PEUVENT être présentes dans WEN :
Vecteur aléatoire
Résumé de message

6.14 Informations d'établissement de liaison (SLI, *Set-Link-Info*)

Le message de contrôle Set-Link-Info est envoyé par un LCCE pour convoyer des informations de changement d'état de liaison ou de circuit concernant le circuit associé à la session L2TP. Par exemple, si PPP renégocie LCP à un LNS ou entre un LAC et un système distant, ou si un circuit virtuel de relais de trame transmis passe à l'état Actif ou Inactif à un LAC, un message SLI DEVRAIT être envoyé pour indiquer l'événement. Les détails précis sur le moment d'envoi de SLI, sur les AVP spécifique du type de PW qui doivent être présentes, et comment ces AVP devraient être interprétées par l'homologue receveur sortent du domaine d'application du présent document. Ces détails devraient être décrits dans les documents associés spécifiques de pseudo filaire qui exigent l'utilisation de ce message.

Les AVP suivantes DOIVENT être présentes dans les SLI :
Type de message
Identifiant de session locale
Identifiant de session distante

Les AVP suivantes PEUVENT être présentes dans les SLI :
Vecteur aléatoire
Résumé de message
État de circuit

6.15 Accusé de réception explicite (ACK)

Le message Accusé de réception explicite (ACK) n'est utilisé que pour accuser réception d'un ou de messages sur la connexion de contrôle (par exemple, pour mettre à jour les valeurs Ns et Nr).La réception de ce message ne déclanche aucun événement pour l'automate à états du protocole L2TP.

Un message reçu sans aucune AVP (incluant l'AVP Type de message) est appelé un message de corps de longueur zéro (ZLB, *Zero Length Body*) et assure la même fonction que l'accusé de réception explicite. Les messages ZLB ne sont permis que lorsque l'authentification de message de contrôle définie au paragraphe 4.3 n'est pas activée.

Les AVP suivantes PEUVENT être présentes dans le message ACK :
Type de message
Résumé de message

7. Automates à états de connexion de contrôle

Les tableaux d'état définis dans cette section gouvernent les échanges de messages de contrôle définis à la Section 6. Les tableaux sont définis pour passer des appels entrants et des appels sortants, ainsi que pour l'initialisation de la connexion de contrôle elle-même. Les tableaux d'état ne codent pas la fin de temporisation ni le comportement de retransmission, car ceci est traité dans le mécanisme sous-jacent de livraison fiable de message de contrôle (voir au paragraphe 4.2).

7.1 AVP mal formées et messages de contrôle

La réception d'un message de contrôle invalide ou mal formé irrécupérable DEVRAIT être enregistrée dans un journal d'événements approprié et la connexion de contrôle éliminée pour s'assurer de la récupération à un état connu. La connexion de contrôle peut alors être redémarrée par l'initiateur.

Un message de contrôle invalide est défini comme (1) un message qui contient un type de message marqué comme obligatoire (voir au paragraphe 5.4.1) mais qui est inconnu de la mise en œuvre, ou (2) un message de contrôle qui est reçu dans le mauvais état.

Des exemples de messages de contrôle mal formés incluent (1) un message qui a une valeur invalide dans son en-tête, (2) un message qui contient une AVP de format incorrect ou dont la valeur est hors gamme, et (3) un message auquel manque une AVP exigée. Un message de contrôle avec un en-tête mal formé DOIT être éliminé.

Lorsque possible, une AVP mal formée devrait être traitée comme une AVP non reconnue (voir au paragraphe 5.2). Donc, une tentative d'inspection du bit M DEVRAIT être faite pour déterminer l'importance de l'AVP mal formée, et donc, la sévérité de la malformation pour le message de contrôle entier. Si le bit M peut être raisonnablement inspecté au sein de l'AVP mal formée et si il est déterminé qu'il doit être établi, comme avec une AVP non reconnue, la session ou connexion de contrôle associée

DOIT alors être fermée. Si le bit M est inspecté et si on trouve qu'il est à 0, l'AVP DOIT être ignorée (en supposant bien sûr que la malformation de l'AVP est récupérable).

Cette politique ne doit pas être considérée comme licence d'envoyer des AVP mal formées, mais plutôt comme un guide de la façon de traiter un message formaté de façon impropre si on en reçoit. Il est impossible de faire la liste de toutes les malformations potentielles d'un certain message et de donner un avis pour chacune. Un exemple de situation d'AVP mal formée qui devrait être récupérable est si l'AVP Vitesse de connexion en réception est reçue avec une longueur de 10 plutôt que 14, ce qui implique que les bits par seconde de la vitesse de connexion sont formatés sur 4 octets plutôt que 8. Si l'AVP n'a pas son bit M établi (comme ce devrait normalement être le cas) cette condition n'est pas considérée comme catastrophique. À ce titre, le message de contrôle devrait être accepté bien que l'AVP ne soit pas présente (mais un message d'erreur local peut être enregistré).

Dans plusieurs cas des tableaux qui suivent, un message de protocole est envoyé, et ensuite un "nettoyage" se produit. Noter que, sans considération de l'initiateur de la destruction de la connexion de contrôle, il doit être permis au mécanisme de livraison fiable de fonctionner (voir au paragraphe 4.2) avant de détruire la connexion de contrôle. Cela permet que les messages de gestion de connexion de contrôle management soient fiablement livrés à l'homologue.

L'Appendice B.1 contient un exemple d'établissement verrouillé de connexion de contrôle.

7.2 État de la connexion de contrôle

Le protocole L2TP de connexion de contrôle ne fait pas de distinction entre les deux LCCE mais fait la distinction entre le générateur et le receveur. L'homologue générateur est celui qui le premier initie l'établissement de la connexion de contrôle. (Dans une situation de nœud, c'est le vainqueur du départage.) Comme le LAC ou le LNS peut être le générateur, une collision peut se produire. Voir l'AVP Départage de connexion de contrôle au paragraphe 5.4.3 qui décrit cela et sa résolution.

État	**Événement**	**Action**	**Nouvel état**
repos	Demande d'ouverture locale	Envoyer SCCRQ	wait-ctl-reply
repos	Reçoit SCCRQ, acceptable	Envoyer SCCRP	wait-ctl-conn
repos	Reçoit SCCRQ, non acceptable	Envoyer StopCCN, nettoyer	repos
repos	Reçoit SCCRP	Envoyer StopCCN, nettoyer	repos
repos	Reçoit SCCCN	Envoyer StopCCN, nettoyer	repos
wait-ctl-reply	Reçoit SCCRP, acceptable	Envoyer SCCCN, envoyer événement control-conn ouvert aux sessions en attente	établi
wait-ctl-reply	Reçoit SCCRP, non acceptable	Envoyer StopCCN, nettoyer	repos
wait-ctl-reply	Reçoit SCCRQ, perd le départage, SCCRQ acceptable	Envoyer SCCRP, nettoyer la connexion perdante	wait-ctl-conn
wait-ctl-reply	Reçoit SCCRQ, perd le départage, SCCRQ non acceptable	Envoyer StopCCN, nettoyer la connexion perdante	repos
wait-ctl-reply	Reçoit SCCRQ, gagne le départage	Envoyer StopCCN pour la connexion perdante	wait-ctl-reply
wait-ctl-reply	Reçoit SCCCN acceptable	Envoyer StopCCN, nettoyer	repos
wait-ctl-conn	Reçoit SCCCN non acceptable,	Envoyer événement control-conn ouvert aux sessions en attente	établi
wait-ctl-conn	Reçoit SCCCN, non acceptable	Envoyer StopCCN, nettoyer	repos
wait-ctl-conn	Reçoit SCCRQ, SCCRP	Envoyer StopCCN, nettoyer	repos
établi	Demande d'ouverture locale (nouvel appel)	Envoyer événement control-conn ouvert aux sessions en attente	établi
établi	Événement de clôture administrative de la connexion de contrôle	Envoyer StopCCN, nettoyer	repos
établi	Reçoit SCCRQ, SCCRP, SCCCN	Envoyer StopCCN, nettoyer	repos
repos, wait-ctl-reply, wait-ctl-conn, établi	Reçoit StopCCN	Nettoyer	repos

Les états associés à un LCCE pour l'établissement de connexion de contrôle sont comme suit :

repos *(idle)*
L'initiateur et le receveur commencent tous deux dans cet état. Un initiateur transmet une SCCRQ, tandis que le receveur reste dans l'état repos jusqu'à recevoir une SCCRQ.

wait-ctl-reply *(attente de réponse de connexion de contrôle)*

Le générateur vérifie pour voir si une autre connexion a été demandée par le même homologue, et si oui, traite la situation de collision décrite au paragraphe 5.4.3.

wait-ctl-conn *(attente de connexion de contrôle)*
Attente d'une SCCCN. Si la SCCCN est valide, la connexion de contrôle est établie ; autrement, elle est supprimée (en envoyant un StopCCN avec le code approprié de résultat et/ou d'erreur).

établi *(established)*
Une connexion établie peut être terminée par une condition locale ou par la réception d'un StopCCN. Dans le cas d'une terminaison locale, le générateur DOIT envoyer un StopCCN et nettoyer la connexion de contrôle. Si le générateur reçoit un StopCCN, il DOIT aussi nettoyer la connexion de contrôle.

7.3 Appels entrants

Une ICRQ est générée par un LCCE, normalement en réponse à un appel entrant ou un événement local. Une fois que le LCCE a envoyé la ICRQ, il attend une réponse de l'homologue. Cependant, il peut choisir de retarder l'établissement de l'appel (par exemple, en répondant à l'appel, en activant le circuit) jusqu'à ce que l'homologue ait indiqué par une ICRP qu'il va accepter l'appel. L'homologue peut choisir de ne pas accepter l'appel si, par exemple, il y a des ressources insuffisantes pour traiter une session supplémentaire.

Si l'homologue choisit d'accepter l'appel, il répond par une ICRP. Lorsque le LCCE local reçoit la ICRP, il tente d'établir l'appel. Un message final d'appel connecté, la ICCN, est envoyé du LCCE local à l'homologue pour indiquer que les états d'appel des deux LCCE devraient entrer dans l'état établi. Si l'appel est terminé avant que l'homologue puisse l'accepter, une CDN est envoyée par le LCCE local pour indiquer cette condition.

Lorsque un appel passe à l'état "déconnecté" ou "arrêté", l'appel est normalement éliminé, et le LCCE local envoie une CDN. De façon similaire, si l'homologue souhaite éliminer un appel, il envoie une CDN et termine sa session.

7.3.1 États d'envoyeur ICRQ

État	Événement	Action	Nouvel état
repos	Signal d'appel ou prêt à recevoir une connexion entrante	Initier l'ouverture locale de la connexion de contrôle	wait-control-conn
repos	Reçoit ICCN, ICRP, CDN	Nettoyer	repos
wait-control-conn	Abandon de la ligne porteuse ou demande de clôture locale	Nettoyer	repos
wait-control-conn	Ouverture de la connexion de contrôle	Envoyer ICRQ	attente réponse
attente réponse	Reçoit ICRP, acceptable	Envoyer ICCN	établi
attente réponse	Reçoit ICRP, non acceptable	Envoyer CDN, nettoyer	repos
attente réponse	Reçoit ICRQ, perd le départage	Traiter comme receveur ICRQ (§ 7.3.2)	repos
attente réponse	Reçoit ICRQ, gagne le départage	Envoyer CDN pour la session perdante	attente réponse
attente réponse	Reçoit CDN, ICCN	Nettoyer	repos
attente réponse	Demande de clôture locale	Envoyer CDN, nettoyer	repos
établi	Reçoit CDN	Nettoyer	repos
établi	Reçoit ICRQ, ICRP, ICCN	Envoyer CDN, nettoyer	repos
établi	Demande de clôture locale	Envoyer CDN, nettoyer	repos

Les états associés à l'envoyeur d'ICRQ sont comme suit :

repos
Le LCCE détecte un appel entrant sur une de ses interfaces (par exemple, une ligne RTPC analogique sonne, ou un PVC ATM est provisionné) ou un événement local survient. Le LCCE initie son automate d'établissement de connexion de contrôle et passe à un état d'attente de confirmation de l'existence d'une connexion de contrôle.

wait-control-conn
Dans cet état, la session attend soit que la connexion de contrôle soit ouverte, soit la vérification que la connexion de contrôle est déjà ouverte. Une fois qu'une indication que la connexion de contrôle est ouverte a été reçue, les messages de contrôle de session peuvent être échangés. Le premier de ces messages est la ICRQ.

attente réponse *(wait-reply)*
L'envoyeur de la ICRQ reçoit soit (1) une CDN qui indique que l'homologue ne veut pas accepter l'appel (erreur générale ou

n'accepte pas) et repasse à l'état repos, soit (2) une ICRP qui indique que l'appel est accepté. Dans ce cas, le LCCE envoie une ICCN et passe à l'état établi.

établi
Des données sont échangées durant la session. L'appel peut être supprimé par une des situations suivantes :
+ Un événement sur l'interface connectée : le LCCE envoie une CDN.
+ Réception d'une CDN : le LCCE nettoie tout, déconnectant l'appel.
+ Une cause locale : le LCCE envoie une CDN.

7.3.2 États du receveur ICRQ

État	Événement	Action	Nouvel état
repos	Reçoit ICRQ, acceptable	Envoyer ICRP	attente connexion
repos	Reçoit ICRQ, non acceptable	Envoyer CDN, nettoyer	repos
repos	Reçoit ICRP	Envoyer CDN nettoyer	repos
repos	Reçoit ICCN	Nettoyer tout	repos
attente connexion	Reçoit ICCN, acceptable	Préparer pour données	établi
attente connexion	Reçoit ICCN, non acceptable	Envoyer CDN, nettoyer	repos
attente connexion	Reçoit ICRQ, ICRP	Envoyer CDN, nettoyer	repos
repos, wait-connect, établi	Reçoit CDN	Nettoyer tout	repos
attente connexion, établi	Demande de clôture locale	Envoyer CDN, nettoyer	repos
établi	Reçoit ICRQ, ICRP, ICCN	Envoyer CDN, nettoyer	repos

Les états associés au receveur de l'ICRQ sont comme suit :
repos
Une ICRQ est reçue. Si la demande n'est pas acceptable, une CDN est renvoyée à l'homologue LCCE, et le LCCE local reste à l'état repos. Si la ICRQ est acceptable, une ICRP est envoyée. La session passe à l'état attente de connexion *(wait-connect)*.
attente de connexion *(wait-connect)*
Le LCCE local attend une ICCN de l'homologue. À réception de l'ICCN, le LCCE local passe à l'état établi.
établi
La session est terminée soit par l'envoi d'une CDN soit par la réception d'une CDN de l'homologue. Le nettoyage suit sur les deux côtés sans considération de qui est l'initiateur.

7.4 Appels sortants

Les appels sortants donnent pour instruction au LAC de passer un appel. Il y a trois messages pour les appels sortants : OCRQ, OCRP, et OCCN. Un LCCE envoie d'abord une OCRQ à un LAC pour demander un appel sortant. Le LAC DOIT répondre à l'OCRQ par une OCRP une fois qu'il a déterminé que les facilités appropriées existent pour passer l'appel et que l'appel est administrativement autorisé. Une fois l'appel sortant connecté, le LAC envoie une OCCN à l'homologue pour indiquer le résultat final de la tentative d'appel.

7.4.1 États de l'envoyeur OCRQ

État	Événement	Action	Nouvel état
repos	Demande d'ouverture locale	Initier ouverture locale connexion de contrôle	wait-control-conn
repos	Reçoit OCCN, OCRP	Nettoyer tout	repos
wait-control-conn	ouverture connexion de contrôle	Envoyer OCRQ	attente réponse
attente réponse	Reçoit OCRP, acceptable	aucune	attente connexion
attente réponse	Reçoit OCRP, non acceptable	Envoyer CDN, nettoyer	repos
attente réponse	Reçoit OCCN	Envoyer CDN, nettoyer	repos
attente réponse	Reçoit OCRQ, perd départage	Traiter comme receveur OCRQ (§ 7.4.2)	repos
attente réponse	Reçoit OCRQ, gagne départage	Envoyer CDN pour session perdante	attente réponse
attente connexion	Reçoit OCCN	aucune	établi
attente connexion	Reçoit OCRQ, OCRP	Envoyer CDN, nettoyer	repos
repos, attente réponse, attente connexion établi	Reçoit CDN	Nettoyer tout	repos
établi	Reçoit OCRQ, OCRP, OCCN	Envoyer CDN, nettoyer	repos
attente réponse, attente connexion, établi	Demande de clôture locale	Envoyer CDN, nettoyer	repos
wait-control-conn	Demande de clôture locale	Nettoyer tout	repos

Les états associés à l'envoyeur de l'OCRQ sont comme suit :

repos, wait-control-conn *(attente connexion de contrôle)*

Lorsque une demande d'appel sortant est initiée, une connexion de contrôle est créée comme décrit ci-dessus, si elle n'est pas déjà présente. Une fois la connexion de contrôle établie, une OCRQ est envoyée au LAC, et la session passe à l'état attente de réponse *(wait-reply)*.

wait-reply

Si une CDN est reçue, la session est nettoyée et retourne à l'état repos. Si une OCRP est reçue, l'appel est en cours, et la session passe à l'état attente connexion *(wait-connect)*.

wait-connect

Si une CDN est reçue, la session est nettoyée et retourne à l'état repos. Si une OCCN est reçue, l'appel a réussi, et la session peut maintenant échanger des données.

établi

Si une CDN est reçue, la session est nettoyée et retourne à l'état repos. Autrement, si le LCCE choisit de terminer la session, il envoie une CDN au LAC, nettoie la session, et la passe à l'état repos.

7.4.2 États du receveur OCRQ (LAC)

État	Événement	Action	Nouvel état
repos	Reçoit OCRQ, acceptable	Envoyer OCRP, établir l'appel	wait-cs-answer
repos	Reçoit OCRQ, non acceptable	Envoyer CDN, nettoyer	repos
repos	Reçoit OCRP	Envoyer CDN, nettoyer	repos
repos	Reçoit OCCN, CDN	Nettoyer tout	repos
wait-cs-answer	Échec de l'établissement d'appel	Envoyer OCCN	établi réussi
wait-cs-answer	Appel établi	Envoyer CDN, nettoyer	repos
wait-cs-answer	Reçoit OCRQ, OCRP, OCCN	Envoyer CDN, nettoyer	repos
établi	Reçoit OCRQ, OCRP, OCCN	Envoyer CDN, nettoyer	repos
wait-cs-answer établie	Reçoit CDN	Nettoyer tout	repos
wait-cs-answer établie,	Demande de clôture locale	Envoyer CDN, nettoyer	repos

Les états associés au LAC pour les appels sortants sont comme suit :

repos

Si l'OCRQ est reçue avec une erreur, répondre par une CDN. Autrement, passer l'appel, envoyer une OCRP, et passer à l'état wait-cs-answer *(attente de la réponse de la commutation de circuit)*.

wait-cs-answer

Si l'appel n'est pas établi ou si une temporisateur arrive à expiration pendant l'attente de l'établissement de l'appel, envoyer une CDN avec la condition d'erreur appropriée, et passer à l'état repos. Si une connexion par commutation de circuit est établie, envoyer une OCCN indiquant le succès, et passer à l'état établi.

établi

Si le LAC reçoit une CDN de l'homologue, l'appel DOIT être libéré via les mécanismes appropriés, et la session être nettoyée. Si l'appel est déconnecté parce que le circuit passe à un état "déconnecté" ou "arrêté", le LAC DOIT envoyer une CDN à l'homologue et retourner à l'état repos.

7.5 Terminaison d'une connexion de contrôle

La terminaison d'une connexion de contrôle consiste en ce que l'homologue produit une StopCCN. L'envoyeur de ce message DEVRAIT attendre l'accusé de réception de ce message pendant un cycle complet de retransmission de message de contrôle (par exemple, 1 + 2 + 4 + 8 ... secondes) avant de libérer les informations de contrôle associées à la connexion de contrôle. Le receveur de ce message devrait envoyer un accusé de réception du message à l'homologue, puis libérer les informations de contrôle associées

Quand libérer une connexion de contrôle est un problème de mise en œuvre et n'est pas spécifié dans le présent document. Une mise en œuvre peut utiliser la politique qui lui paraît appropriée pour déterminer quand libérer une connexion de contrôle. Certaines mises en œuvre peuvent laisser une connexion de contrôle ouverte pendant une certaine période ou parfois indéfiniment après la libération de la dernière session de cette connexion de contrôle. D'autres peuvent choisir de déconnecter la connexion de contrôle immédiatement après la déconnexion du dernier appel sur la connexion de contrôle.

8. Considérations sur la sécurité

La présente section traite de certaines des questions de sécurité que L2TP rencontre dans son fonctionnement.

8.1 Sécurité du point de connexion de contrôle et du message

Si un secret partagé (mot de passe) existe entre deux LCCE, il peut être utilisé pour effectuer une authentification mutuelle entre les deux LCCE, et construire une vérification d'authentification et d'intégrité des messages de contrôle L2TP arrivants. Le mécanisme fourni par L2TPv3 est décrit au paragraphe 4.3 et dans la définition des AVP Résumé de message et Nom occasionnel d'authentification de message de contrôle au paragraphe 5.4.1.

Ce mécanisme de sécurité de message de contrôle assure (1) l'authentification mutuelle de point d'extrémité, et (2) la vérification de l'intégrité et de l'authenticité des messages de contrôle individuels. L'authentification mutuelle de point d'extrémité assure qu'une connexion de contrôle L2TPv3 n'est établie qu'entre deux points d'extrémité qui sont configurés avec le mot de passe approprié. La vérification d'intégrité des messages de contrôle individuels protège de la corruption accidentelle ou intentionnelle des paquets (c'est-à-dire, celle causée par une imitation de message de contrôle ou une attaque par interposition).

Le secret partagé qui est utilisé pour toutes les connexions de contrôle, messages de contrôle, et les caractéristiques de sécurité d'AVP définies dans le présent document n'a jamais besoin d'être envoyé en clair entre les points d'extrémité de tunnel L2TP.

8.2 Paquet de données contrefaits

La contrefaçon de paquet pour tout type de protocole de réseau privé virtuel (VPN, *Virtual Private Network*) est un souci particulier car l'insertion de paquets contrefaits construits avec soin dans le réseau VPN de transit pourrait résulter en une violation de la séparation des trafics du VPN, et à la fuite des données dans un VPN client. Ceci est compliqué par le fait qu'il peut être particulièrement difficile à l'opérateur du VPN d'être même informé qu'il est devenu un point de transit dans ou entre des VPN clients.

L2TPv3 assure la séparation des trafics pour ses VPN via un identifiant de session de 32 bits dans l'en-tête des données L2TPv3. Lorsque il est présent, le mouchard L2TPv3 (décrit au paragraphe 4.1) fournit une vérification supplémentaire pour s'assurer qu'un paquet arrivant est destiné à la session identifiée. Donc, l'utilisation d'un mouchard avec l'identifiant de session fournit une garantie supplémentaire que la recherche de l'identifiant de session a été effectuée correctement et que l'identifiant de session lui-même n'a pas été corrompu dans le transit.

En présence d'une attaque aveugle de paquets contrefaits, le mouchard peut aussi assurer la sécurité contre des fuites de trames par inadvertance dans un VPN client. Pour illustrer le type de sécurité fournie dans ce cas, considérons la comparaison de la validation d'un mouchard de 64 bits dans l'en-tête L2TPv3 avec l'admission de paquets qui correspondent à une certaine paire d'adresses IP de source et destination. La validation de la paire d'adresses IP de source et destination et la validation du mouchard consistent toutes deux en une vérification rapide des informations d'en-tête en clair sur tous les paquets arrivants. Cependant, comme L2TPv3 utilise sa propre valeur, il supprime l'exigence de tenir une liste des adresses IP permises ou refusées, et plus encore, de garder la connaissance des adresses IP permises contre des pirates qui peuvent les obtenir et les imiter. De plus, il est bien plus facile de changer un mouchard L2TP compromis qu'une adresse IP "compromise", et une valeur cryptographiquement aléatoire [RFC1750] a bien moins de chances d'être découverte par des attaques en force brute qu'une adresse IP.

Pour la protection contre les attaques en force brute, aveugles, d'insertion, un mouchard de 64 bits DOIT être utilisé avec toutes les sessions. Un mouchard de 32 bits est vulnérable à une recherche en force brute à des débits de paquets élevés, et à ce titre, ne devrait pas être considéré comme une barrière efficace contre les attaques d'insertion aveugles bien qu'il soit toujours utile comme vérification supplémentaire d'une recherche réussie d'identifiant de session). Le mouchard ne donne aucune protection contre une attaque par interposition sophistiquée qui peut renifler et corréler les données capturées entre les nœuds pour les utiliser dans une attaque coordonnée.

L'AVP Mouchard alloué est utilisée pour signaler la valeur et la taille du mouchard qui doit être présent dans tous les paquets de données pour une session. Chaque mouchard alloué DOIT être choisi de façon aléatoire chiffrée [RFC1750] de telle façon qu'une série de mouchards alloués ne fournisse aucune indication sur ce que sera un futur mouchard.

Le mouchard L2TPv3 ne doit pas être considéré comme un substitut de la sécurité telle que fournie par IPsec lors du fonctionnement sur un réseau ouvert ou qui n'est pas de confiance où les paquets peuvent être reniflés, décodés, et corrélés pour être utilisés dans une attaque coordonnée. Voir au paragraphe 4.1.3 plus d'informations sur L2TP sur IPsec.

9. Considérations d'internationalisation

Les AVP Nom d'hôte et Nom du fabricant ne sont pas internationalisées. L'AVP Nom du fabricant, bien que destinée à être lisible par l'homme, semblerait entrer dans la catégorie des "noms visibles mondialement" [RFC2277] et est donc représentée en US-ASCII.

Si (1) un LCCE ne signifie pas une préférence de langage par l'inclusion d'une AVP Langage préféré (voir au paragraphe 5.4.3) dans la SCCRQ ou la SCCRP, (2) l'AVP Langage préféré n'est pas reconnue, ou (3) le langage demandé n'est pas pris en charge par le LCCE homologue, le langage par défaut de la [RFC2277] DOIT être utilisé pour toutes les chaînes internationalisées envoyées par l'homologue.

10. Considérations relatives à l'IANA

Le présent document définit un certain nombre de numéros "magiques" qui seront enregistrés par l'IANA. Cette section explique les critères utilisés par l'IANA pour allouer des numéros supplémentaires dans chacune de ces listes. Les paragraphes qui suivent décrivent la politique d'allocation pour les espaces de noms définis ailleurs dans ce document. Les paragraphes 10.1 à 10.3 sont des demandes de nouvelles valeurs déjà gérées par l'IANA conformément à la [RFC3438]. Les paragraphes restants sont pour les nouveaux registres qui ont été ajoutés au registre L2TP existant et sont en conséquence tenus par l'IANA.

10.1 Paires valeur/attribut de message de contrôle

Cet espace de numéros est géré par l'IANA selon la [RFC3438]. Voici un résumé des nouvelles AVP :

Paires valeur/attribut de message de contrôle

Type d'attribut	Description
58	AVP identifiant de fabricant étendu
59	Résumé de message
60	Identifiant de routeur
61	Identifiant alloué de connexion de contrôle
62	Liste des capacités pseudo filaires
63	Identifiant de session locale
64	Identifiant de session distante
65	Mouchard alloué
66	Identifiant d'extrémité distante
68	Type pseudo filaire
69	Sous couche spécifique de couche 2
70	Séquençage de données
71	État de circuit
72	Langage préféré
73	Nom occasionnel d'authentification de message de contrôle
74	Vitesse de connexion en émission
75	Vitesse de connexion en réception

10.2 Valeurs d'AVP de type de message

Cet espace de numéros est géré par l'IANA selon la [RFC3438]. Il y a un nouveau type de message, défini au paragraphe 3.1, qui a été alloué pour cette spécification:

AVP Type de message (type d'attribut 0)	Valeur
Gestion de la connexion de contrôle	20 (ACK) Accusé de réception explicite

10.3 Valeurs d'AVP de code de résultat

Cet espace de numéros est géré par l'IANA selon la [RFC3438]. Les nouvelles valeurs de code de résultat pour le message CDN sont définies au paragraphe 5.4. Voici un résumé :

Valeurs	AVP de code de résultat (type d'attribut 1)
	Codes d'erreur générale
13	Session non établie due à la perte du départage (L2TPv3).
14	Session non établie due à un type de PW non pris en charge (L2TPv3).
15	Session non établie, séquençage requis sans sous couche spécifique de couche 2 valide (L2TPv3).
16	Erreur d'automate à états finis ou fin de temporisation.

10.4 Bits d'en-tête d'AVP

Ce nouveau registre sera tenu par l'IANA.

Bits de tête de l'en-tête d'AVP L2TP
Il y a six bits au début de l'en-tête d'AVP L2TP. Les nouveaux bits sont alloués par action de normalisation de la [RFC2434] :
Bit 0 - Obligatoire (bit M)
Bit 1 - Caché (bit H)
Bit 2 - Réservé
Bit 3 - Réservé
Bit 4 - Réservé
Bit 5 - Réservé

10.5 Bits d'en-tête de message de contrôle L2TP

Ce nouveau registre sera tenu par l'IANA.

Bits de tête de l'en-tête de message de contrôle L2TP
Il y a 12 bits au début de l'en-tête du message de contrôle L2TP. Les bits réservés devraient seulement être définis par action de normalisation [RFC2434] :
Bit 0 - Type de message (T bit)
Bit 1 - Champ Longueur si présent (bit L)
Bit 2 - Réservé
Bit 3 - Réservé
Bit 4 - Numéros de séquence présente (bit S)
Bit 5 - Réservé
Bit 6 - Champ Décalage si présent [RFC2661]
Bit 7 - Bit Priorité (bit P) [RFC2661]
Bit 8 - Réservé
Bit 9 - Réservé
Bit 10 - Réservé
Bit 11 - Réservé

10.6 Types pseudo filaire

Ce nouveau registre sera tenu par l'IANA, il n'y a aucune valeur allouée dans le présent document.

Types pseudo filaires L2TPv3
Le type pseudo filaire (type PW, voir au paragraphe 5.4) est une valeur de 2 octets utilisée dans l'AVP Type pseudo filaire et l'AVP Liste de capacités pseudo filaires définies au paragraphe 5.4.3. Les valeurs 0 à32 767 sont allouées par revue d'expert [RFC2434], tandis que les valeurs de 32 768 à 65 535 sont allouées selon la politique du premier arrivé, premier servi [RFC2434]. Il n'y a pas de type spécifique pseudo filaire alloué dans ce document. Chaque document spécifique de pseudo filaire doit allouer ses propres types PW à partir de l'IANA comme nécessaire.

10.7 Bits État de circuit

Ce nouveau registre sera tenu par l'IANA.

Bits État de circuit
Le champ État de circuit est un gabarit de 16 bits, dont les deux bits de moindre poids sont alloués. Des bits supplémentaires peuvent être alloués par consensus de l'IETF [RFC2434].

Bit 14 - Nouveau (bit N)
Bit 15 - Actif (bit A)

10.8 Bits de la sous couche par défaut spécifique de couche 2

Ce nouveau registre sera tenu par l'IANA.

Bits de sous couche spécifique de couche 2 par défaut
La sous couche spécifique de couche 2 par défaut contient 8 bits dans la portion de moindre poids de l'en-tête. Les bits réservés peuvent être alloués par consensus de l'IETF [RFC2434].

Bit 0 - Réservé
Bit 1 - Séquence (bit S)
Bit 2 - Réservé
Bit 3 - Réservé
Bit 4 - Réservé
Bit 5 - Réservé
Bit 6 - Réservé
Bit 7 - Réservé

10.9 Type de sous couche spécifique de couche 2

Ce nouveau registre sera tenu par l'IANA.

Type de sous couche spécifique de couche 2
Le type de sous couche spécifique de couche 2 est un entier non signé de deux octets. Des valeurs supplémentaires peuvent être allouées par revue d'expert [RFC2434].

0 - Pas de sous couche spécifique de couche 2
1 - Sous couche spécifique de couche 2 par défaut présente

10.10 Niveau de séquençage des données

Ce nouveau registre sera tenu par l'IANA.
Niveau de séquençage des données
Le niveau de séquençage des données est un entier non signé de deux octets. Des valeurs supplémentaires peuvent être allouées par revue d'expert [RFC2434].

0 – Aucun paquet de données entrant n'exige de séquençage.
1 – Seuls les paquets de données non IP exigent le séquençage.
2 – Tous les paquets de données entrants exigent le séquençage.

11. Références

11.1 Références normatives

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

[RFC2277] H. Alvestrand, "Politique de l'IETF en matière de jeux de caractères et de langages", BCP 18, janvier 1998.

[RFC2434] T. Narten et H. Alvestrand, "Lignes directrices pour la rédaction d'une section Considérations relatives à l'IANA dans les RFC", BCP 26, octobre, 1998. *(Rendue obsolète par la* RFC5226)

[RFC2473] A. Conta, S. Deering, "Spécification du tunnelage générique de paquet dans IPv6", décembre 1998. *(P.S.)*

[RFC2661] W. Townsley et autres, "Protocole de tunnelage de couche 2 "L2TP"", *(P.S.)*

[RFC2865] C. Rigney et autres, "Service d'authentification à distance de l'utilisateur appelant (RADIUS)", juin 2000. (*MàJ par* RFC2868, RFC3575, RFC5080) *(D.S.)*

[RFC3066] H. Alvestrand, "Étiquettes pour l'identification des langues", BCP 47, janvier 2001. *(Obsolète, voir la* RFC4646.)

[RFC3193] B. Patel et autres, "Sécuriser L2TP avec IPsec", novembre 2001. *(P.S.)*

[RFC3438] W. Townsley, "Mise à jour des considérations de l'IANA sur le protocole de tunnelage de couche deux (L2TP)", décembre 2002. (BCP0068)

[RFC3629] F. Yergeau, "UTF-8, un format de transformation de la norme ISO 10646", STD 63, novembre 2003.

11.2 Références pour information

[RFC1034] P. Mockapetris, "Noms de domaines - Concepts et facilités", STD 13, novembre 1987.

[RFC1191] J. Mogul et S. Deering, "Découverte de la MTU de chemin", novembre 1990.

[RFC1321] R. Rivest, "Algorithme de résumé de message MD5", avril 1992. *(Information)*

[RFC1661] W. Simpson, éditeur, "Protocole point à point (PPP)", STD 51, juillet 1994. *(MàJ par la* RFC2153)

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

[RFC1750] D. Eastlake 3rd et autres, "Recommandations d'aléa pour la sécurité", décembre 1994. *(Info., remplacée par la* RFC4086)

[RFC1958] B. Carpenter, éd., "Principes de l'architecture de l'Internet", juin 1996. (*MàJ par* RFC3439) *(Information)*

[RFC1981] J. McCann, S. Deering, J. Mogul, "Découverte de la MTU de chemin pour IP version 6", août 1996. *(D.S.)*

[RFC2072] H. Berkowitz, "Guide du dénumérotage des routeurs", janvier 1997. (*MàJ par* RFC4192) *(Information)*

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

[RFC2341] A. Valencia, M. Littlewood, T. Kolar, "Protocole de transmission de couche 2 "L2F" de Cisco", mai 1998. *(Historique)*

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

[RFC2581] M. Alman, V. Paxson et W. Stevens, "Contrôle d'encombrement avec TCP", avril 1999. *(Obsolète, voir* RFC5681)

[RFC2637] K. Hamzeh, G. Pall, W. Verthein, J. Taarud, W. Little et G. Zorn, "Protocole de tunnelage point à point (PPTP)", juillet 1999.

[RFC2732] R. Hinden, B. Carpenter et L. Masinter, "Format pour les adresses littérales IPv6 dans les URL", décembre 1999. *(Obsolète, voir* RFC3986) *(P.S.)*

[RFC2809] B. Aboba, G. Zorn, "Mise en œuvre du tunnelage obligatoire L2TP via RADIUS", avril 2000. *(Information)*

[RFC3070] V. Rawat et autres, "Protocole de tunnelage de couche 2 (L2TP) sur relais de trame", février 2001. *(P.S.)*

[RFC3335] T. Harding, R. Drummond, C. Shih, "Échange de données d'affaire sécurisées d'homologue à homologue fondé sur MIME sur l'Internet", septembre 2002. *(P.S.)*

[KPS] Kaufman, C., Perlman, R., et Speciner, M., "Network Security: Private Communications in a Public World", Prentice Hall, mars 1995, ISBN 0-13-061466-1.

[STEVENS] Stevens, W. Richard, "TCP/IP Illustrated, Volume I: The Protocols", Addison-Wesley Publishing Company, Inc., mars 1996, ISBN 0-201-63346-9.

12. Remerciements

De nombreuses constructions du protocole ont été définies à l'origine dans la RFC 2661, "L2TPv2", ainsi que le texte du début du présent document. Les auteurs de la RFC 2661 sont W. Townsley, A. Valencia, A. Rubens, G. Pall, G. Zorn et B. Palter. Le concept de base de L2TP et beaucoup des constructions du protocole ont été adoptées dans L2F [RFC2341] et PPTP [RFC2637]. Les auteurs de ces versions sont A. Valencia, M. Littlewood, T. Kolar, K. Hamzeh, G. Pall, W. Verthein, J. Taarud, W. Little, et G. Zorn. Danny Mcpherson et Suhail Nanji ont publié la première version de "Type de service L2TP", qui définissait l'utilisation de L2TP pour le tunnelage de divers types de charge utile L2 (initialement, Ethernet et relais de trame). L'équipe qui a partagé la RFC 2661 en ce document de base et son document PPP d'accompagnement était composée de Ignacio Goyret, Jed Lau, Bill Palter, Mark Townsley, et Madhvi Verma. Skip Booth a aussi fourni une révision très utile et des

commentaires. Certaines constructions de L2TPv3 se fondaient en partie sur l'interface de transport universelle (UTI, *Universal Transport Interface*) qui a été conçue à l'origine par Peter Lothberg et Tony Bates. Stewart Bryant et Simon Barber ont fourni de précieux apports pour L2TPv3 sur en-tête IP. Juha Heinanen a fait une utile relecture des premières étapes de cet effort. Jan Vilhuber, Scott Fluhrer, David McGrew, Scott Wainner, Skip Booth et Maria Dos Santos ont contribué au mécanisme d'authentification de message de contrôle ainsi qu'aux discussions générales sur la sécurité. James Carlson, Thomas Narten, Maria Dos Santos, Steven Bellovin, Ted Hardie, et Pekka Savola ont fait une très utile relecture des versions finales du texte. Russ Housley a fait une utile relecture et des commentaires sur la sécurité, en particulier par rapport au mécanisme d'authentification de message de contrôle. Pekka Savola a contribué au bon alignement sur IPv6 et a inspiré beaucoup du paragraphe 4.1.4 sur la fragmentation. En plus de son influence originale et de sa qualité de co-auteur de la RFC 2661, Glen Zorn a aidé à saisir correctement toutes les références de langage et de caractères dans le présent document. Un certain nombre de gens ont fourni de précieux apports et efforts pour la RFC 2661, sur laquelle se fonde le présent document. John Bray, Greg Burns, Rich Garrett, Don Grosser, Matt Holdrege, Terry Johnson, Dory Leifer, et Rich Shea ont fourni de précieux apports et révisions au 43ème IETF à Orlando, FL, qui ont conduit à améliorer la lisibilité globale et la clarté de la RFC 2661. Thomas Narten a fourni un grand nombre de critiques sur le formatage. Il a écrit la première version de la section des considérations relatives à l'IANA. Dory Leifer a apporté de précieuses précisions à la définition du protocole de L2TP et contribué à l'édition des premières versions conduisant à la RFC 2661. Steve Cobb et Evan Caves ont redessiné les tableaux des automates à états. Barney Wolff a fourni de nombreux apports à la conception du mécanisme original d'authentification de point d'extrémité.

Appendice A Démarrage lent du contrôle et évitement d'encombrement

Bien que chaque côté ait indiqué la taille maximum de sa fenêtre de réception, il est recommandé qu'une méthode de démarrage lent et d'évitement d'encombrement soit utilisée pour transmettre les paquets de contrôle. Les méthodes décrites ici se fondent sur l'algorithme TCP d'évitement d'encombrement décrit au paragraphe 21.6 de "TCP/IP Illustrated", Volume I, par W. Richard Stevens [STEVENS] (cet algorithme est aussi décrit dans la [RFC2581]).

Démarrage lent et évitement d'encombrement utilisent plusieurs variables. La fenêtre d'encombrement (CWND) définit le nombre de paquets qu'un envoyeur peut envoyer avant d'attendre un accusé de réception. La taille de CWND s'étend et se contracte comme décrit plus loin. Noter cependant que CWND ne doit jamais excéder la taille de fenêtre annoncée obtenue dans l'AVP Fenêtre de réception. (Dans le texte qui suit, on suppose que toute augmentation sera limitée par la taille de la fenêtre de réception.) La variable SSTHRESH *(seuil de démarrage lent)* détermine quand l'envoyeur passe du démarrage lent à l'évitement d'encombrement. Le démarrage lent est utilisé lorsque CWND est inférieur à SSHTRESH.

Un envoyeur commence dans la phase de démarrage lent. CWND est initialisé à un paquet, et SSHTRESH est initialisé à la fenêtre annoncée (obtenue de l'AVP Fenêtre de réception). L'envoyeur transmet ensuite un paquet et attend son accusé de réception (explicite ou porté). Lorsque l'accusé de réception est reçu, la fenêtre d'encombrement est incrémentée de un à deux. Durant le démarrage lent, CWND est augmenté de un paquet chaque fois qu'un ACK (message ACK explicite ou porté) est reçu. Augmenter CWND de un à chaque ACK a pour effet de doubler CWND à chaque aller retour, résultant en un accroissement exponentiel. Lorsque la valeur de CWND atteint SSHTRESH, la phase démarrage lent se termine et la phase d'évitement d'encombrement commence.

Durant l'évitement d'encombrement, CWND s'accroît plus lentement. Précisément, il augmente de 1/CWND à chaque nouvel ACK reçu. C'est-à-dire que CWND augmente d'un paquet après la réception de nouveaux ACK. L'expansion de la fenêtre durant la phase d'évitement d'encombrement est effectivement linéaire, avec CWND augmentant de un paquet à chaque aller retour.

Lorsque l'encombrement se produit (indiqué par le déclanchement d'une retransmission) une moitié de la CWND est sauvegardée dans SSTHRESH, et CWND est réglé à un. L'envoyeur entre alors à nouveau dans la phase de démarrage lent.

Appendice B Exemples de message de contrôle

B.1 Établissement d'une connexion de contrôle Lock-Step

Dans cet exemple, un LCCE établit une connexion de contrôle, chaque côté impliqué dans l'échange alternant l'envoi des messages. Cet exemple montre explicitement l'accusé de réception final avec un message ACK. Une solution de remplacement serait le portage de l'accusé de réception au sein d'un message envoyé en réponse à l'ICRQ ou OCRQ qui va probablement suivre de la part du côté qui a initié la connexion de contrôle.

```
     LCCE A          LCCE B
SCCRQ        ->
Nr: 0, Ns: 0
                     <-   SCCRP
                     Nr: 1, Ns: 0
SCCCN        ->
Nr: 1, Ns: 1
                     <-    ACK
                     Nr: 2, Ns: 1
```

B.2 Paquet perdu avec retransmission

Une connexion de contrôle existante a une nouvelle session demandée par le LCCE A. L'ICRP est perdue et doit être retransmise par le LCCE B. Noter que la perte de la ICRP a deux effets : non seulement cela empêche l'automate à états de niveau supérieur de progresser, mais cela empêche aussi le LCCE A de voir en temps utile l'accusé de réception de niveau inférieur de son ICRQ.

```
                LCCE A                              LCCE B
ICRQ       >
Nr: 1, Ns: 2
(paquet perdu)                                <-    ICRP
                                              Nr: 3, Ns: 1
      (pause ; le temporisateur du LCCE A a démarré le premier et se termine le premier)
ICRQ     ->
Nr: 1, Ns: 2
(Réalisant qu'il a déjà vu ce paquet, le LCCE B élimine le paquet et envoie un message ACK)
                                              <-    ACK
Nr: 3, Ns: 2
                    (Le temporisateur du LCCE B arrive à expiration)
                                              <-    ICRP
                                              Nr: 3, Ns: 1
ICCN     ->
Nr: 2, Ns: 3
                                              <-    ACK
                                              Nr: 4, Ns: 2
```

Appendice C Traitement des numéros de séquence

La sous couche spécifique de couche 2 par défaut, définie au paragraphe 4.6, donne un champ de 24 bits pour le séquençage des paquets de données au sein d'une session L2TP. Les paquets de données L2TP ne sont jamais retransmis, de sorte que cette séquence n'est utilisée que pour détecter l'ordre des paquets, les paquets dupliqués, ou les paquets perdus.

Le champ de 24 bits Numéro de séquence de la sous couche spécifique de couche 2 par défaut contient un numéro de séquence de paquets pour la session associée. Chaque paquet de données séquencé envoyé doit contenir le numéro de séquence, incrémenté de un, du précédent paquet séquencé envoyé sur une certaine session L2TP. À réception, tout paquet qui a un numéro de séquence égal ou supérieur à celui du paquet actuellement attendu (le dernier paquet en ordre reçu plus un) devrait être considéré comme "nouveau" et accepté. tous les autres paquets sont considérés comme "périmés" ou "dupliqués" et éliminés. Noter que l'espace de numéros de séquence de 24 bits inclut zéro comme numéro de séquence valide (à ce titre, il peut être mis en œuvre si on le désire avec un compteur de 32 bits masqué). Toutes les nouvelles sessions DOIVENT commencer par envoyer les numéros de séquences à zéro.

De plus grands ou plus petits champs de numéros de séquence sont possibles avec L2TP si on utilise un autre format pour la sous couche spécifique de couche 2 par défaut que celui défini dans le présent document. Bien que 24 bits puisse être adéquat dans de nombreuses circonstances, un plus large espace de numéros de séquence sera moins susceptible de poser des problèmes de retour à zéro du numéro de séquence pour les session à très haut débit de données sur de longues périodes. Les recommandations de traitement des numéros de séquence ci-dessous devraient tenir pour des champs de numéro de séquence de toutes tailles.

Quand on détecte si le numéro de séquence d'un paquet est "supérieur" ou "inférieur" à la valeur d'un certain numéro de séquence, le retour à zéro du numéro de séquence doit être pris en compte. Ceci se fait normalement en gardant une fenêtre de numéros de séquence au delà de celui actuellement attendu pour déterminer si un paquet est "nouveau" ou non. La taille de la

fenêtre peut se fonder sur le débit de la liaison et l'espace de numéros de séquence et DEVRAIT être configurable avec une valeur par défaut égale à la moitié de la taille de l'espace de numéros disponible (par exemple, 2^(n-1), où n est le nombre de bits disponibles dans le numéro de séquence).

À réception, les paquets qui correspondent exactement au numéro de séquence attendu sont traités immédiatement et le prochain numéro de séquence attendu est incrémenté. Les paquets qui tombent dans la fenêtre de nouveaux paquets peuvent soit être traités immédiatement et le prochain numéro de séquence attendu mis à jour à un de plus que celui reçu dans le nouveau paquet, soit conservés pour un très bref instant dans l'espoir de la réception du ou des paquets manquants. Ce "très bref instant" devrait être configurable, avec une valeur par défaut correspondant à une durée au moins d'un ordre de grandeur inférieure aux périodes de temporisation de retransmission des protocoles de couche supérieure comme TCP.

Pour les paquets en transit typiques déclassés, le simple abandon des paquets déclassés devrait suffire et exige généralement moins de ressources que de réorganiser activement les paquets au sein de L2TP. Une exception est le cas où une paire de fragments de paquets sont retransmis de façon persistante et envoyés déclassés. Par exemple, si un paquet IP a été fragmenté en très petits paquets suivis par un très gros paquet avant d'être tunnelé par L2TP, il est possible (bien qu'incorrect) que les deux paquets L2TP résultants puissent être déclassés par le PSN en transit entre les nœuds L2TP. Si des numéros de séquence sont appliqués au nœud receveur sans aucune mise en mémoire tampon des paquets déclassés, le paquet IP fragmenté peut ne jamais arriver à destination. Il peut être bon de noter ici que cette condition est vraie pour tout mécanisme de tunnelage de paquets IP qui inclut des vérifications de numéro de séquence à réception (c'est-à-dire, GRE [RFC2890]).

L'utilisation d'un niveau de séquençage de données (voir au paragraphe 5.4.3) de 1 (seuls les paquets de données non IP exigent le séquençage) permet aux paquets de données IP qui sont tunnelés par L2TP de ne pas utiliser de numéros de séquence, tandis qu'on utilise des numéros de séquence et on applique l'ordre des paquets pour tous les paquets de données non IP restants. Selon les exigences de la couche de liaison qui est tunnelée et des données du réseau qui traversent la liaison de données, ceci est suffisant dans de nombreux cas pour appliquer l'ordre des paquets sur les trames qui l'exigent (comme les messages de contrôle de liaison des données de bout en bout) et pas sur les paquets IP qui sont connus pour être résilients au reclassement de paquet.

Si un grand nombre de paquets (c'est-à-dire, plus d'une fenêtre de nouveaux paquets) sont éliminés à cause d'une grosse panne ou de la perte de l'état des numéros de séquence sur un côté de la connexion (peut-être au titre d'un rétablissement du plan de transmission ou d'une reprise sur échec sur un nœud de secours) il est possible qu'un grand nombre de paquets soient envoyés dans l'ordre, mais soient mal détectés par l'homologue comme étant déclassés. Ceci peut être généralement caractérisé pour une taille de fenêtre, w, un espace de numéros de séquence, s, et un nombre de paquets perdus dans le transit entre des points d'extrémité L2TP, p, comme suit :

Si s > p > w, alors des paquets supplémentaires (s - p) qui auraient autrement été reçus dans l'ordre, vont être incorrectement classés comme n'étant pas dans l'ordre et éliminés. Donc, pour un espace de numéros de séquence, s = 128, une taille de fenêtre, w = 64, et un nombre de paquets perdus, p = 70 ; 128 - 70 = 58 paquets supplémentaires vont être éliminés après la panne jusqu'à ce que le numéro de séquence revienne au prochain numéro de séquence en cours attendu.

Pour atténuer cette perte de paquets supplémentaire, on DOIT inspecter les numéros de séquence des paquets éliminés à cause d'un classement comme "périmés" et rétablir en conséquence le numéro de séquence attendu. Cela peut se faire en comptant le nombre de paquets "périmés" éliminés qui étaient en séquence entre eux et, quand on atteint un certain seuil, en rétablissant le prochain numéro de séquence attendu à celui vu dans les paquets de données qui arrivent. Les horodatages des paquets peuvent aussi être utilisés comme indicateur pour rétablir le numéro de séquence attendu en détectant la période sur laquelle les paquets "périmés" ont été reçus en séquence. Le seuil idéal va varier selon la vitesse de la liaison, l'espace de numéros de séquence, et la tolérance de la liaison aux paquets déclassés, et DOIT être configurable.

Adresse des éditeurs

Jed Lau
cisco Systems
170 W. Tasman Drive
San Jose, CA 95134
mél : jedlau@cisco.com

W. Mark Townsley
cisco Systems
mél : mark@townsley.net

Ignacio Goyret
Lucent Technologies
mél : igoyret@lucent.com

Déclaration de droits de reproduction

pour ce qui est mentionné ci-après, les auteurs conservent tous leurs droits.

Le présent document et les informations qui y sont contenues sont fournis sur une base "EN L'ÉTAT" et le contributeur, l'organisation qu'il ou elle représente ou qui le/la finance (s'il en est), la INTERNET SOCIETY et la INTERNET ENGINEERING TASK FORCE déclinent toutes garanties, exprimées ou implicites, y compris mais non limitées à toute garantie que l'utilisation des informations ci-encloses ne violent aucun droit ou aucune garantie implicite de commercialisation ou d'aptitude à un objet particulier.

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 faits 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 .

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