

Groupe de travail Réseau
Request for Comments : 1989
RFC rendue obsolète : 1333
Catégorie : En cours de normalisation

W. Simpson
Daydreamer
août 1996
Traduction Claude Brière de L'Isle

Surveillance de qualité de liaison PPP

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 protocole point à point (PPP) [RFC1661] donne une méthode standard pour le transport des datagrammes multi protocoles sur des liaisons point à point.

PPP définit aussi un protocole de commande de liaison extensible, qui permet la négociation d'un protocole de qualité pour une surveillance continue de la viabilité de la liaison.

Le présent document définit un protocole pour générer des rapports de qualité de la liaison.

Table des Matières

1.	Introduction.....
2.	Surveillance de la qualité de liaison.....
2.1	Motivations du concept.....
2.2	Compteurs.....
2.3	Comptages des paquets et octets.....
2.4	Traitements.....
2.5	Format d'option de configuration.....
2.6	Format de paquet.....
2.7	Transmission des rapports.....
2.8	Calculs.....
2.9	Détection de défaillance.....
2.10	Suggestions de politique.....
	Considérations pour la sécurité.....
	Remerciements.....
	Références.....
	Adresse du président.....
	Adresse de l'auteur.....

1. Introduction

Dans le but d'établir des communications sur une liaison point à point, chaque extrémité de la liaison PPP doit d'abord envoyer des paquets du protocole de contrôle des liaisons (LCP) pour configurer la liaison de données durant la phase d'établissement de la liaison. Durant les phases d'authentification et de protocole de couche réseau, la liaison peut subir des vérifications pour déterminer si sa qualité est suffisante pour le fonctionnement. Cette vérification est complètement facultative.

Si une mise en œuvre désire que son homologue utilise un protocole spécifique de surveillance de qualité de liaison, elle DOIT alors négocier l'utilisation de ce protocole à l'aide de l'option Configuration de protocole de qualité durant la phase d'établissement de la liaison.

Le mécanisme de négociation est indépendant dans chaque direction. Cependant, si l'homologue est d'accord pour envoyer des paquets de protocole de qualité, elle DOIT traiter correctement de tels paquets à réception, même si elle ne demande pas de tels paquets ou si elle ne met pas en œuvre une politique de surveillance.

2. Surveillance de la qualité de liaison

Les liaisons de communication de données sont rarement parfaites. Des paquets peuvent être abandonnés ou corrompus pour diverses raisons (bruit de ligne, défaillances d'équipements, débordement de mémoires tampons, etc.). Parfois, il est souhaitable de déterminer quand, et à quelle fréquence, la ligne abandonne des données. Par exemple, des routeurs peuvent vouloir admettre temporairement qu'un autre chemin prenne la préséance. Une mise en œuvre peut aussi avoir l'option de se déconnecter et de passer sur une autre liaison. Le processus de détermination des pertes de donnée est appelé "surveillance de qualité de liaison".

2.1 Motivations du concept

Il y a de nombreuses façons différentes de mesurer la qualité de la liaison, et encore plus de façons d'y réagir. Plutôt que de spécifier un seul schéma, la surveillance de qualité de liaison se divise en un "mécanisme" et une "politique". PPP spécifie complètement le "mécanisme" de la surveillance de qualité de liaison en définissant le paquet de rapport de qualité de liaison (LQR, *Link Quality Report*) et en spécifiant une procédure pour l'utiliser.

PPP NE SPÉCIFIE PAS une "politique" de surveillance de qualité de liaison – comment juger de la qualité de la liaison ou de ce qu'il faut faire quand elle est inadéquate. Ceci est laissé à la décision de la mise en œuvre, et peut être différent à chaque extrémité de la liaison. Les mises en œuvre ont la liberté d'expérimenter des politiques de qualité de liaison variées, et sont même invitées à le faire. La spécification du mécanisme de surveillance de qualité de liaison assure que deux mises en œuvre qui ont des politiques différentes peuvent communiquer et interopérer.

Pour permettre la mise en œuvre de politiques souples, le mécanisme de surveillance de qualité de liaison PPP mesure les pertes de données en unités de paquets, d'octets, et de rapports de qualité de liaison. Chaque mesure est faite séparément pour chaque moitié de la liaison, à la fois en entrée et en sortie. Toutes les mesures sont communiquées aux deux extrémités de la liaison, de sorte que chaque extrémité de la liaison puisse mettre en œuvre sa propre politique de qualité de liaison pour les deux liaisons, entrante et sortante.

Finalement, le protocole de surveillance de qualité de liaison est conçu pour pouvoir être mis en œuvre sur de nombreuses sortes de systèmes différents. Bien qu'il soit courant de mettre en œuvre PPP (et en particulier la surveillance de qualité de liaison) comme un seul processus logiciel, des mises en œuvre multi processus avec un support matériel sont aussi envisagées. Le mécanisme de surveillance de qualité de liaison de PPP apporte une définition soignée du format du paquet de rapport de qualité de liaison et spécifie les points de référence pour toutes les mesures de transmission et de réception des données.

2.2 Compteurs

Chaque mise en œuvre de surveillance de la qualité des liaisons tient le compte du nombre de paquets et d'octets transmis et bien reçus, et transmet périodiquement ces informations à son homologue dans un paquet de rapport de qualité de liaison.

Ces compteurs sont similaires aux numéros de séquence ; ils augmentent constamment pour donner une indication "relative" du nombre de paquets et d'octets communiqués à travers la liaison sortante. En comparant les valeurs des rapports de qualité de liaison (LQR) successifs, un receveur de LQR peut calculer le "delta" de paquets et d'octets communiqués avec succès sur la liaison. La comparaison de ces nombres absolus donne alors une indication de la qualité d'une liaison. Les nombres relatifs sont transmis plutôt que les absolus parce que cela simplifie grandement la synchronisation de la liaison.

Le rapport de qualité de liaison utilise les compteurs d'interface définis dans la MIB-II SNMP [RFC1212]. Ces compteurs ne sont pas initialisés à une valeur particulière lorsque le LCP entre dans la phase Établissement.

De plus, le rapport de qualité de liaison exige la mise en œuvre des trois compteurs suivants, non signés, à accroissement croissant monotone qui se conforment aux exigences de type et de taille de la MIB des compteurs SNMP [RFC1155].

OutLQRs

OutLQRs est un compteur de 32 bits qui s'augmente d'un à chaque paquet de rapport de qualité de liaison transmis. Ce compteur DOIT être mis à zéro lorsque le LCP entre dans la phase Établissement, et NE DOIT PAS être remis à zéro avant que le LCP ne quitte la phase Terminaison. Ce compteur est incrémenté avant qu'il soit inséré dans le paquet LQR.

InLQRs

InLQRs est un compteur de 32 bits qui s'augmente de un pour chaque paquet de rapport de qualité de liaison reçu. Ce compteur DOIT être mis à zéro quand le LCP entre dans la phase Établissement, et NE DOIT PAS être remis à zéro tant que le LCP n'a pas quitté la phase Terminaison. Ce compteur est incrémenté avant qu'il soit inséré (d'une façon qui est à la

discrétion de la mise en œuvre) dans le paquet LQR.

InGoodOctets

InGoodOctets est un compteur de 32 bits qui s'augmente du nombre d'octets contenus dans chaque paquet de couche Liaison des données bien reçu. À la différence de la MIB ifInOctets, les octets pour les trames qui sont comptés dans ifInDiscards et ifInErrors NE DOIVENT PAS être comptés. Ce compteur PEUT être remis à n'importe quelle valeur initiale lorsque le LCP entre dans la phase Établissement, mais NE DOIT PAS être remis à la valeur initiale tant que le LCP n'a pas quitté la phase de Terminaison.

2.3 Comptages des paquets et octets

L'intention des compteurs est de fournir une indication de la quantité d'informations qui passent sur la liaison, plutôt qu'une mesure réelle de la bande passante totale utilisée. La présente spécification est conçue pour donner le même compte dans des circonstances variées, comme lorsque un appareil séparé donne des mécanismes de tramages et d'échappement invisibles de la mise en œuvre, ou qu'un convertisseur de synchrone en asynchrone sur la liaison fait les changements entre les mécanismes.

Tous les octets qui sont inclus dans le calcul de FCS (*Frame Check Sequence*, séquence de contrôle de trame) DOIVENT être comptés, y compris l'en-tête de paquet, le champ Informations, et tout bourrage. Les octets de FCS DOIVENT aussi être comptés et un octet fanion par trame DOIT être compté. Tous les autres octets (comme de séquences de fanion supplémentaires, et les bits ou octets d'échappement) NE DOIVENT PAS être comptés.

Lors de l'insertion du paquet et des comptes d'octet dans le LQR, les comptes DOIVENT inclure les valeurs attendues ou le LQR lui-même.

2.4 Traitements

Le mécanisme de surveillance de qualité de liaison PPP est décrit en utilisant un modèle de "traitement logique". Comme on le montre ci-dessous, il y a cinq processus logiques dupliqués à chaque extrémité de la liaison duplex.

```

+-----+ +-----+ +-----+ Sortie
|         |-->| Mux  |-->| Tx  |=====>
|Gestion. | +-----+ +-----+
| de      |
| liaison | +-----+ +-----+ Entrée
|         |<--| Demux |<--| Rx  |<=====
+-----+ +-----+ +-----+

```

Gestionnaire de liaison

Le processus de gestionnaire de liaison transmet et reçoit les rapports de qualité de liaison, et met en œuvre la politique de qualité de liaison désirée. Les paquets de LQR sont transmis à débit constant, qui est négocié par l'option LCP de configuration de protocole de qualité.

Mux

Le processus Mux multiplexe les paquets à partir de divers protocoles (par exemple, LCP, IP, XNS, etc.) en un seul flux séquentiel et priorisé de paquets. Les paquets de rapport de qualité de liaison DOIVENT recevoir la plus haute priorité possible pour assurer que les informations de qualité de la liaison sont communiquées à temps.

Tx

Le processus Tx entretient les compteurs de la MIB ifOutUniPackets et ifOutOctets, et le compteur interne OutLQRs, qui sont utilisés pour mesurer la quantité de données qui est transmise sur la liaison sortante. Lorsque Tx traite un paquet de rapport de qualité de liaison, il insère les valeurs de ces compteurs dans les champs PeerOut... correspondants du paquet. Le processus Tx DOIT suivre le processus Mux afin que les paquets soient comptés dans l'ordre de transmission à la liaison.

Rx

Le processus Rx entretient les compteurs de la MIB ifInUniPackets, ifInDiscards, ifInErrors et IfInOctets, et les compteurs internes InLQRs et InGoodOctets, qui sont utilisés pour mesurer la quantité de données qui sont reçues par la liaison sortante. Lorsque Rx traite un paquet de rapport de qualité de liaison, il insère les valeurs de ces compteurs dans les champs SaveIn... correspondants du paquet (d'une façon qui est à la discrétion de la mise en œuvre).

Demux

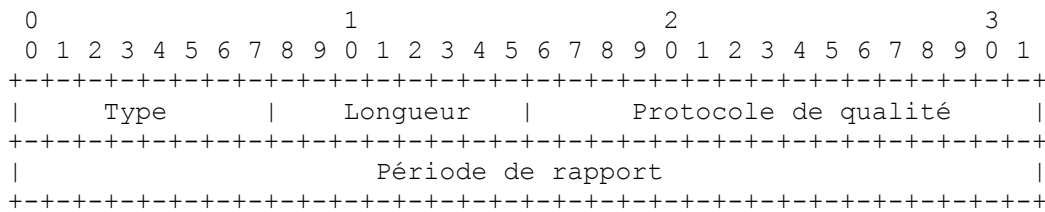
Le processus Demux démultiplexe les paquets pour les divers protocoles. Le processus Demux DOIT suivre le processus Rx afin que les paquets soient comptés dans l'ordre où ils sont reçus de la liaison.

2.5 Format d'option de configuration

Description

Les mises en œuvre DOIVENT être prêtes à recevoir l'option Configuration de protocole de qualité pour le rapport de qualité de liaison. Cependant, la négociation n'est pas requise. La négociation n'est nécessaire que lorsque la mise en œuvre souhaite s'assurer que l'homologue transmet les rapports de qualité de liaison et non d'autres protocoles de qualité, ou autrement pour empêcher l'homologue de tenir son propre temporisateur, ou encore d'établir un délai maximum entre les transmissions de rapports de qualité de liaison.

On montre ci-après un schéma du format de l'option de configuration de protocole de qualité pour négocier le rapport de qualité de liaison. Les champs sont transmis de gauche à droite.



Type : 4

Longueur : 8

Protocole de qualité : c025 (hex) pour le rapport de qualité de liaison

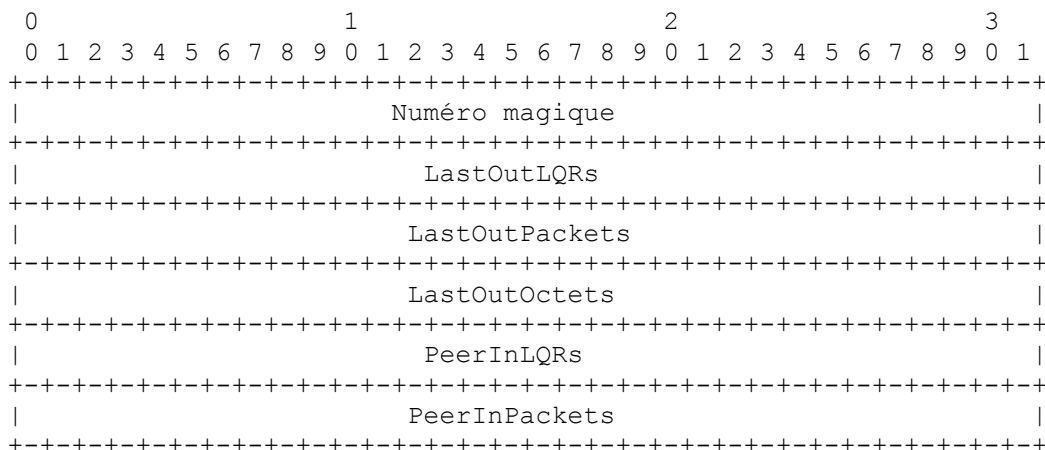
Période de rapport

Le champ Période de rapport est de quatre octets et indique la durée maximum en centièmes de secondes entre la transmission des paquets. L'homologue PEUT transmettre les paquets à un débit plus élevé que celui qui a été négocié.

Une valeur de zéro indique que l'homologue n'a pas besoin d'entretenir un temporisateur. L'homologue peut à la place générer un LQR immédiatement après avoir reçu un LQR. Une valeur de zéro DOIT recevoir un accusé de non réception (NACK) de la part de l'homologue avec une valeur appropriée différente de zéro quand l'homologue a envoyé ou va envoyer un paquet Demande de configuration qui contient l'option Configuration de protocole de qualité pour le rapport de qualité de liaison avec une période de rapport de zéro.

2.6 Format de paquet

Exactement un paquet de rapport de qualité de liaison est encapsulé dans le champ Information des trames de couche de liaison des données de PPP où le champ Protocole indique le type hex c025 (Rapport de qualité de liaison). Un résumé du format du paquet LQR est présenté ci-dessous. Les noms des champs se rapportent au receveur du paquet, car c'est le receveur qui a demandé le paquet dans l'option Configuration. Les champs sont transmis de gauche à droite.



```

|                               PeerInDiscards                               |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               PeerInErrors                               |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               PeerInOctets                               |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               PeerOutLQRs                               |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               PeerOutPackets                           |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               PeerOutOctets                             |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

Les champs suivants ne sont en fait pas transmis sur la liaison entrante. Ils sont plutôt ajoutés logiquement (d'une façon qui dépend de la mise en œuvre) au paquet par le processus Rx de la mise en œuvre.

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               SaveInLQRs                               |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               SaveInPackets                           |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               SaveInDiscards                           |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               SaveInErrors                             |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               SaveInOctets                             |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

Numéro magique

Le champ Numéro magique fait quatre octets et aide à détecter les liaisons qui sont en condition de rétro bouclage. Sauf modifié par une option Configuration, le numéro magique DOIT être transmis à zéro et DOIT être ignoré à réception. Si les numéros magiques ont été négociés, les paquets LQR entrants DEVRAIENT être vérifiés pour s'assurer que l'extrémité locale ne voit pas son propre numéro magique et qu'il y a donc une liaison en rétro bouclage. Voir des explications complémentaires sous l'option Configuration de numéro magique.

LastOutLQRs

Le champ LastOutLQRs fait quatre octets, et à la transmission, il est copié des PeerOutLQRs les plus récemment reçus.

LastOutPackets

Le champ LastOutPackets fait quatre octets, et à la transmission, il est copié des PeerOutPackets les plus récemment reçus.

LastOutOctets

Le champ LastOutOctets fait quatre octets, et à la transmission, il est copié des PeerOutOctets les plus récemment reçus.

PeerInLQRs

Le champ PeerInLQRs fait quatre octets, et à la transmission, il est copié des SaveInLQRs les plus récemment reçus. Chaque fois que le champ PeerInLQRs se trouve être à zéro, les champs LastOut... sont indéterminés, et les champs PeerIn... contiennent les valeurs initiales pour l'homologue.

PeerInPackets

Le champ PeerInPackets fait quatre octets, et à la transmission, il est copié des SaveInPackets les plus récemment reçus.

PeerInDiscards

Le champ PeerInDiscards fait quatre octets, et à la transmission, il est copié des SaveInDiscards les plus récemment reçus.

PeerInErrors

Le champ PeerInErrors fait quatre octets, et à la transmission, il est copié des SaveInErrors les plus récemment reçus..

PeerInOctets

Le champ PeerInOctets fait quatre octets, et à la transmission, il est copié des SaveInOctets les plus récemment reçus.

PeerOutLQRs

Le champ PeerOutLQRs fait quatre octets, et à la transmission, il est copié des OutLQRs. Ce numéro DOIT inclure ce LQR.

PeerOutPackets

Le champ PeerOutPackets fait quatre octets, et à la transmission, il est copié des ifOutUniPackets et ifOutNUniPackets de la MIB actuelle. Ce numéro DOIT inclure ce LQR.

PeerOutOctets

Le champ PeerOutOctets fait quatre octets, et à la transmission, il est copié du ifOutOctets de la MIB actuelle. Ce numéro DOIT inclure ce LQR.

SaveInLQRs

Le champ SaveInLQRs fait quatre octets, et il est copié à réception de InLQRs. Ce numéro DOIT inclure ce LQR.

SaveInPackets

Le champ SaveInPackets fait quatre octets, et à réception, il est copié des ifInUniPackets et ifInNUniPackets de la MIB actuelle. Ce numéro DOIT inclure ce LQR.

SaveInDiscards

Le champ SaveInDiscards fait quatre octets, et à réception, il est copié de ifInDiscards de la MIB actuelle. Ce numéro DOIT inclure ce LQR.

SaveInErrors

Le champ SaveInErrors fait quatre octets, et à réception, il est copié de ifInErrors de la MIB actuelle. Ce numéro DOIT inclure ce LQR.

SaveInOctets

Le champ SaveInOctets fait quatre octets, et à réception, il est copié du InGoodOctets actuel. Ce numéro DOIT inclure ce LQR.

Noter que InGoodOctets n'est pas le même que le compteur ifInOctets de la MIB, car InGoodOctets n'inclut pas d'octets pour les paquets qui sont éliminés ou erronés.

2.7 Transmission des rapports

Lorsque le protocole de contrôle de liaison PPP a atteint l'état Ouvert, le processus de surveillance de qualité de liaison PEUT commencer à envoyer les rapports de qualité de liaison. Si un message Protocole rejeté est reçu qui spécifie un paquet LQR, le processus LQM DOIT cesser d'envoyer des LQR.

Normalement, le LQR est transmis lorsque le temporisateur LQR expire pour la liaison. Si aucun temporisateur de LQR n'est utilisé, un LQR est généré à réception d'un LQR entrant. Le processus de négociation assure qu'au moins un côté de la liaison utilise un temporisateur de LQR.

De plus, un LQR est généré chaque fois que deux LQR successifs sont reçus qui ont la même valeur PeerInLQRs. Cela peut indiquer qu'un LQR a été manqué, ou que la mise en œuvre envoie à un rythme significativement plus lent que l'homologue, ou que l'homologue a accéléré sa génération de LQR pour mieux quantifier les erreurs sur la liaison.

Chaque fois qu'un LQR est envoyé, le temporisateur de LQR DOIT être redémarré.

2.8 Calculs

Chaque fois qu'est reçu un paquet de rapport de qualité de liaison de la liaison entrante, le gestionnaire de liaison peut comparer les champs associés. Les champs du LQR précédent peuvent être soustraits des valeurs de LQR actuelles pour obtenir un "delta" absolu, qui permet de comparer les changements vus par chaque extrémité de la liaison.

Si le champ PeerInLQRs reçu est de zéro, les champs LastOut... sont indéterminés, et les champs PeerIn... contiennent les valeurs initiales pour l'homologue. Aucun calcul utilisant ces champs ne peut être effectué à ce moment.

Note pour la mise en œuvre :

Les compteurs suivants reviennent à zéro lorsque la valeur maximum est atteinte. Il faut veiller à s'assurer que les calculs de delta corrects sont effectués à ce moment.

Le champ LastOutLQRs peut être comparé directement avec les champs PeerInLQRs pour déterminer combien de LQR sortants ont été perdus.

Le champ LastOutLQRs peut être comparé directement au compteur OutLQRs pour déterminer combien de LQR sortants sont encore en chemin.

Le changement dans PeerInPackets peut être comparé à celui de LastOutPackets pour déterminer le nombre de paquets perdus sur la liaison sortante.

Le changement dans PeerInOctets peut être comparé avec celui dans LastOutOctets pour déterminer le nombre d'octets perdus sur la liaison sortante.

Le changement dans SaveInPackets peut être comparé à celui de PeerOutPackets pour déterminer le nombre de paquets perdus sur la liaison entrante.

Le changement dans SaveInOctets peut être comparé à celui de PeerOutOctets pour déterminer le nombre d'octets perdus sur la liaison entrante.

Le changement dans les champs PeerInDiscards et PeerInErrors peut être utilisé pour déterminer si la perte de paquets est due à l'encombrement chez l'homologue plutôt qu'à une défaillance physique de la liaison.

2.9 Détection de défaillance

Lorsque la liaison fonctionne bien dans les deux directions de la liaison, le LQR est superflu. L'intervalle de temps maximum pour la transmission des LQR DEVRAIT être choisi pour interférer de façon minimale avec le trafic actif.

Lorsque il y a une perte de données mesurable dans l'une ou l'autre direction, si le débit global est adéquat, les conditions ne sont pas assez sévères pour justifier l'abandon de la liaison. L'envoi plus rapide de LQR n'apporte rien, sauf de mesurer les pointes du taux de pertes. L'intervalle de temps DOIT être choisi assez long pour avoir un bon effet de lissage sur les données, tout en étant assez court pour assurer une réponse assez rapide à une défaillance complète.

Lorsque la liaison est bonne en entrée, mais très mauvaise en sortie, les LQR entrants indiquent une forte perte sur le côté sortant de la liaison. L'envoi plus rapide de LQR ne sert à rien parce qu'ils seront probablement perdus sur le chemin vers l'homologue.

Lorsque la liaison est bonne en sortie mais très mauvaise en entrée, les LRQ entrants seront fréquemment perdus. Dans ce cas, les LQR DEVRAIENT être envoyés plus rapidement. Il incombe principalement à l'homologue de prendre une décision de politique bien informée. L'homologue va aussi envoyer des LQR en réponse (dus au champ PeerInLQRs dupliqué) et quelques uns de ces LQR peuvent réussir à arriver.

Lorsque un LQR n'arrive pas dans les délais prévus, ou lorsque le LQR reçu indique que les liaisons sont vraiment mauvaises, au moins un LQR supplémentaire DEVRAIT être envoyé. Une décision algorithmique exige au moins deux intervalles d'aller retour. Le taux de perte pourrait être transitoire, dû à une lourde charge de la liaison, ou à un LQR sortant perdu.

2.10 Suggestions de politique

Les paquets de rapport de qualité de liaison fournissent un mécanisme pour déterminer la qualité de la liaison, mais il appartient à chaque mise en œuvre de décider si la liaison est utilisable. Il est recommandé que cette politique mette en œuvre une certaine quantité d'hystérèse afin que la liaison ne soit pas agitée de soubresauts. Une politique est d'utiliser un algorithme de K parmi N. Dans un tel algorithme, il doit y avoir K succès sur les N dernières périodes pour que la liaison soit considérée comme de bonne qualité.

Les procédures pour récupérer de liaisons de mauvaise qualité ne sont pas spécifiées et peuvent varier d'une mise en œuvre à l'autre. Une approche suggérée est de fermer immédiatement tous les autres protocoles de couche réseau (c'est-à-dire, faire que IPCP transmette une demande de terminaison) mais de continuer à transmettre les rapports de qualité de liaison. Une fois que la qualité de la liaison revient à un niveau acceptable, les protocoles de couche réseau peuvent être reconfigurés.

Considérations pour la sécurité

Les questions de sécurité ne sont pas abordées dans le présent mémoire.

Remerciements

Une partie du texte de ce document est tirée de la RFC1172, par Drew Perkins de la Carnegie Mellon University, et par Russ Hobby de la University of California à Davis.

Des remerciements tout particuliers à Craig Fox (Network Systems), et Karl Fox (Morning Star Technologies), pour les suggestions des concepts fondées sur l'expérience de la mise en œuvre.

Références

[RFC1661] W. Simpson, éditeur, "[Protocole point à point](#) (PPP)", STD 51, juillet 1994. *(MàJ par la RFC 2153)*

[RFC1212] M. Rose et K. McCloghrie, "[Définitions concises de MIB](#)", STD 16, février 1991.

[RFC1155] M. Rose et K. McCloghrie, "Structure et [identification des informations de gestion](#) pour les internets fondés sur TCP/IP", STD 16, mai 1990.

Adresse du président

Le groupe de travail peut être contacté via son actuel président :

Karl Fox
Ascend Communications
3518 Riverside Drive, Suite 101
Columbus, Ohio 43221
mél : karl@ascend.com

Adresse de l'auteur

Les questions sur le présent mémoire peuvent aussi être adressées à :

William Allen Simpson
Daydreamer
Computer Systems Consulting Services
1384 Fontaine
Madison Heights, Michigan 48071
mél Bill.Simpson@um.cc.umich.edu
ou bsimpson@MorningStar.com (de préférence)