

Groupe de travail sur les réseaux
Requête pour Commentaires : 1570
Met à jour : 1548
Catégorie : Standard
Traduction : Yves lescop

W. Simpson, Editor
Daydreamer
janvier 1994

Lycée la croix-rouge - Brest

Extensions LCP du PPP

Statut de ce document

Ce document spécifie un protocole standard d'Internet pour la communauté Internet, et ne sera éprouvé qu'après plusieurs discussions et suggestions. Merci de vous référer à l'édition courante du " Internet Official Protocol Standards " (STD1) pour l'état de standardisation et le statut de ce protocole. La distribution de ce document est illimitée.

Résumé

Le Protocole point à point (PPP) [1] fournit une méthode standard pour transporter les datagrammes multiprotocole sur des liens point à point. La PPP définit un protocole de contrôle de lien extensible (LCP : Link Control Protocol) pour établir, configurer, et tester la connexion de liaison de données. Ce document définit plusieurs dispositifs supplémentaires LCP qui ont été suggérés au cours de ces dernières années.

Ce document est le produit du groupe de travail sur le protocole point à point de l'IETF (Internet Engineering Task Force). Des commentaires devraient être soumis à la liste de diffusion : ietf-ppp@ucdavis.edu.

Sommaire

1	PAQUETS SUPPLÉMENTAIRES DE LCP	2
1.1	IDENTIFICATION	2
1.2	TEMPS-RESTANT	4
2	OPTIONS SUPPLÉMENTAIRES DE CONFIGURATION LCP.....	6
2.1	FCS ALTERNATIVES	6
2.1.1	<i>Considérations LCP</i>	7
2.1.2	<i>FCS Nulle</i>	7
2.2	REPLISSAGE DÉCRIT PAR SOI-MÊME	8
2.3	RAPPEL	10
2.4	TRAMES COMPOSÉES	11
2.4.1	<i>Considérations de LCP</i>	12
A.	IMPLANTATION RAPIDE DU CONTRÔLE DE TRAME (FCS).....	13
A.1.	MÉTHODE DE CALCUL DU FCS 32 BITS	13
	CONSIDÉRATIONS SÉCURITAIRES	15
	RÉFÉRENCES.....	15
	REMERCIEMENTS	15
	ADRESSE DU COMITÉ.....	16
	ADRESSE DE L'ÉDITEUR.....	16

1 Paquets Supplémentaires de LCP

Le format de paquet et les facilités de base sont déjà définis pour LCP [1].

Des valeurs à jour du champ code de LCP sont indiquées dans le RFC "nombres assignés" [2] le plus récent. Ces spécifications concernent les valeurs suivantes :

- 12 Identification
- 13 Temps-Restant

1.1 Identification

Description :

Ce code fournit une méthode pour une implantation permettant de s'identifier à son pair. Ce code pourrait être utilisé pour beaucoup de fonctions diverses, telles que le dépannage du lien, l'application de privilèges, etc....

L'identification est un paquet d'entretien de lien. Des paquets d'identification PEUVENT être

envoyés à tout moment, avant que LCP ait atteint l'état ouvert inclut.

L'expéditeur transmet un paquet LCP avec le champ code réglé à 12 (identification), le champ identificateur positionné, le Nombre magique local (si présent) inséré, et le champ message rempli de n'importe quelles données désirées, mais sans excéder le MRU par défaut minoré de huit.

La réception d'un paquet d'identification provoque l'événement RXR ou RUC. Il n'y a aucune réponse au paquet d'identification.

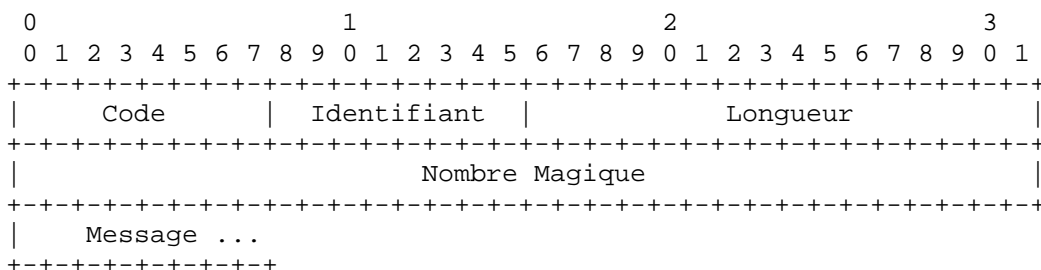
La réception d'un code Rejet pour le paquet d'identification DEVRAIT produire de l'événement RXJ+ (autorisé).

Justification :

Ce dispositif est défini en tant qu'élément de LCP, plutôt que comme protocole séparé de PPP, pour que ses avantages puissent être disponibles pendant l'étape la plus préliminaire possible de la phase d'établissement du lien. Il permet à un opérateur d'apprendre l'identification du pair même lorsque la négociation n'est pas convergente. Des paquets non-LCP ne peuvent pas être envoyés pendant la phase d'établissement du lien.

Ce dispositif est défini comme un code séparé de LCP, plutôt qu'une option de configuration, de sorte que le pair n'ait pas besoin de l'inclure avec d'autres éléments dans des échanges de paquet de configuration, et manipule des valeurs "corrigées" ou "rejet", puisque sa génération est rare et dans une seule direction. Il est recommandé que des paquets d'identification soient envoyés toutes les fois qu'un rejet de configuration est envoyé ou reçu, comme message final quand la négociation ne converge pas, et quand LCP atteint l'état ouvert.

Un récapitulatif du format du paquet d'identification est montré ci-dessous. Les champs sont transmis de gauche à droite.



Code : 12 pour Identification

Identifiant : Le champ identifiant DOIT être changé pour chaque identification émise.

Longueur : > = 8

Nombre Magique : Le champ Nombre Magique est de quatre octets et aide à la détection de liens qui sont dans un état de bouclage. Jusqu'à ce que l'option de configuration du Nombre Magique ait été négociée avec succès, le nombre magique DOIT être transmis en tant que zéro. Voyez l'option de configuration du Nombre Magique pour davantage d'explications.

Message : Le champ message est de zéro octets ou plus, et son contenu dépend de l'implantation. Il est destiné à être lisible par un humain, et NE DOIT PAS affecter l'exécution du protocole. Il est recommandé que le message contienne des caractères ASCII affichables (de 32 à 126 en décimal). Les mécanismes pour l'extension à d'autres jeux de caractères sont l'objet d'une recherche future. La taille est déterminée à partir du champ longueur.

Note de Mise en place :

Le message contiendra habituellement des choses telles que type du matériel de l'émetteur, niveau de révision du logiciel PPP, numéro de série du produit PPP, information de MIB telle que le débit du lien et le nom d'interface, et n'importe quelle autre information que l'expéditeur pense être utile dans la mise au point des connexions. Le format est susceptible d'être différent pour chaque implanteur, de sorte que ceux qui font le cheminement de numéro de série puissent valider leurs nombres. Une mise en place robuste DEVRAIT traiter le message en tant que texte affichable, et DEVRAIT pouvoir recevoir et afficher un message très long.

1.2 Temps-Restant

Description :

Ce code fournit un mécanisme pour informer le pair du temps restant pour cette session.

La nature de cette information est consultative seulement. Il est entendu que seul un côté de la connexion enverra ce paquet (généralement "un serveur d'accès réseau"). On clôture réellement la session par un paquet "Demande de Terminaison".

"Temps-Restant" est un paquet d'entretien du lien. Des paquets "Temps-Restant" peuvent seulement être introduits pendant l'état ouvert du LCP.

L'expéditeur transmet un paquet LCP avec le champ code réglé à 13 (Temps-Restant), le positionnement du champ identifiant, le Nombre Magique local (si présent) inséré, et le champ message rempli de n'importe quelles données désirées, mais sans excéder le MRU du pair minoré de douze.

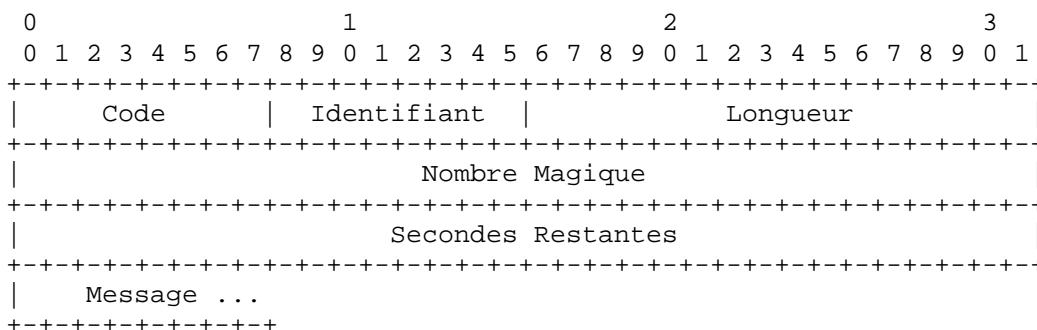
La réception d'un paquet "Temps-Restant" provoque l'événement RXR ou RUC. Il n'y a aucune réponse au paquet "Temps-Restant".

La réception du "Code-Rejet" pour le paquet "Temps-Restant" DEVRAIT produire l'événement RXJ+ (autorisé).

Justification :

Cet avis est défini comme un code séparé de LCP, plutôt qu'une option de configuration, pour que les changements et les messages d'avertissement puissent se produire dynamiquement pendant la session, et ainsi l'information pourrait être déterminée après la phase d'authentification. Typiquement, ce paquet est envoyé quand le lien entre dans la phase de protocole de couche réseau, et à intervalles réguliers durant toute la session, en particulier près de la fin de la session.

Un récapitulatif du format du paquet "Temps-Restant" est montré ci-dessous. Les champs sont transmis de gauche à droite.



Code : 13 pour "Temps-Restant"

Identifiant : Le champ identifiant DOIT être changé pour chaque "Temps-Restant" envoyé.

Longueur : > = 12

Nombre Magique : Le champ Nombre Magique est de quatre octets et aide à la détection de liens qui sont dans un état de bouclage. Jusqu'à ce que l'option de configuration du Nombre Magique ait été négociée avec succès, le nombre magique DOIT être transmis en tant que zéro. Voyez l'option de configuration du Nombre Magique pour davantage d'explications.

Secondes Restantes : Le champ "Secondes Restantes" est de quatre octets et indique le nombre de secondes intégrales restantes pour cette session. Cette valeur non signée de 32 bits est envoyée avec l'octet le plus significatif d'abord. Une valeur de 0xffffffff (tout à 1) indique aucune minuterie, ou "pour toujours".

Message : Le champ message est de zéro octets ou plus, et son contenu dépend de l'implantation. Il est destiné à être lisible par un humain, et NE DOIT PAS affecter l'exécution du protocole. Il est recommandé que le message contienne des caractères ASCII affichables (de 32 à 126 en décimal). Les mécanismes pour l'extension à d'autres jeux de caractères sont l'objet d'une recherche future. La taille est déterminée à partir du champ longueur.

2 Options Supplémentaires de Configuration LCP

Le format d'option de configuration et les options de base sont déjà définis pour LCP [1].

Des valeurs à jour du champ type d'option du LCP sont indiquées dans le RFC "nombres assignés" [2] le plus récent. Ce document concerne les valeurs suivantes :

9	FCS alternatives
10	Remplissage décrit par soi-même
13	Rappel
15	Trames composées

2.1 FCS Alternatives

Description :

Cette option de configuration fournit une méthode pour une implantation permettant d'indiquer un autre format de FCS à envoyer par le pair, ou pour négocier la FCS globalement.

Cette option est négociée séparément dans chaque direction. Cependant, on n'exige pas de l'implantation qu'elle soit capable de produire concurremment une FCS différente de chaque côté du lien.

Les valeurs négociées de FCS entrent en vigueur seulement pendant les phases d'authentification et de protocole couche réseau. Les trames envoyées pendant n'importe quelle autre phase DOIVENT contenir la FCS par défaut.

Un récapitulatif du format d'option de configuration de FCS Alternatives est montré ci-dessous. Les champs sont transmis de gauche à droite.

```

      0                               1                               2
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
  +-----+-----+-----+-----+-----+-----+
  |      Type      |      Longueur      |      Options      |
  +-----+-----+-----+-----+-----+-----+

```

Type : 9

Longueur : 3

Options : Ce champ est d'un octet, et est un composé "ou logique" des valeurs suivantes :

1 FCS nul

- 2 FCS 16 bits CCITT
- 4 FCS 32 bits CCITT

Note de Mise en place :

Pour la plupart des liens en trames PPP HDLC, la FCS par défaut est la FCS 16 bits du CCITT. Quelques techniques de trames et liens à grande vitesse peuvent utiliser un autre format comme FCS par défaut.

2.1.1 Considérations LCP

Le lien peut être sujet à la perte d'état, et le LCP peut renégocier à tout moment. Quand le LCP commence la renégociation ou l'arrêt, il est recommandé que les paquets LCP "Demande-Configuration" ou "Demande-Terminaison" soient envoyés avec la dernière FCS négociée, puis on change alors pour la FCS par défaut, et un paquet LCP dupliqué est envoyé avec la FCS par défaut. Le champ identifiant NE DEVRAIT PAS être incrémenté pour chaque paquet ainsi dupliqué.

À la réception d'un paquet LCP "Demande-Configuration" ou "Demande-Terminaison", la mise en place DOIT permuter vers la FCS par défaut pour la transmission et la réception. Si on reçoit un paquet de demande qui contient un champ identificateur dupliqué, une nouvelle réponse DOIT être produite.

Notes de Mise en place :

L'envoi de deux paquets est seulement nécessaire après que la FCS alternative ait été déjà négociée. Cela n'a pas besoin de se produire pendant les transitions d'état quand il y a une indication normale que la FCS par défaut est effective, comme pour les événements "marche" ou "arrêt". Il est nécessaire d'envoyer deux paquets pour les états Envoi-ACK et ouverts, puisque le pair pourrait croire de manière erronée que le lien s'est ouvert.

Il est possible d'envoyer une FCS 48-bits simple qui est une combinaison des FCS 16 bits et 32 bits. Ceci peut être envoyé au lieu d'envoyer les deux paquets décrits ci-dessus. Nous n'avons pas normalisé ce procédé en raison des soucis de propriété intellectuelle. Si une telle FCS 48-bits est utilisée, elle DOIT seulement être utilisée pour des paquets LCP.

2.1.2 FCS Nulle

La FCS nulle DEVRAIT seulement être utilisée pour les protocoles de couche réseau et de transport qui ont une somme de contrôle de bout en bout disponible, comme TCP/IP, ou UDP/IP avec le total de contrôle autorisé. C'est-à-dire, l'option de FCS nulle DEVRAIT être négociée ainsi qu'une autre option de FCS non nulle dans un environnement hétérogène.

Quand un paquet de configuration (LCP ou NCP) ou d'authentification est envoyé, la FCS DOIT être incluse. Quand un paquet de configuration (LCP ou NCP) ou d'authentification est reçu, la FCS DOIT être vérifiée.

Il y a plusieurs cas à considérer :

FCS nulle seule

L'expéditeur génère la FCS des trames qui exigent la FCS avant d'envoyer la trame.

Quand une trame est reçue, il n'est pas nécessaire de contrôler la FCS avant le démultiplexage. N'importe quelle FCS est traitée comme remplissage.

La réception d'un paquet d'authentification ou de contrôle serait découverte après passage de la trame au démultiplexeur. La vérification de la FCS peut facilement être accomplie en utilisant un des algorithmes de logiciel définis dans "PPP dans les trames HDLC" [3] (FCS de 16 bits) et l'annexe A (FCS de 32 bits).

FCS nulle avec une autre FCS, utilisant le logiciel

C'est similaire au cas ci-dessus.

Les paquets qui exigent la présence d'une FCS (authentification, contrôle, ou Protocoles Réseau avec total de contrôle absent) sont contrôlés en utilisant le logiciel après le démultiplexage. Des paquets qui ne passent pas le test de FCS sont jetés comme d'habitude.

FCS nulle avec une autre FCS, utilisant du matériel

Un indicateur est passé avec la trame, indiquant s'il a passé ou non le contrôle de FCS matériel. La FCS incorrecte DOIT être passée avec le reste des données. La trame NE DOIT PAS être jetée jusqu'après le démultiplexage, et seulement les trames qui exigent la FCS sont jetées.

Chacune des trois formes de FCS (nulle, 16 et 32) peut être utilisée concurremment sur différentes trames en utilisant le logiciel. Ce n'est probablement pas possible avec la plupart des matériels actuels.

2.2 Remplissage décrit par soi-même

Description

Cette option de configuration fournit une méthode pour une mise en place indiquant au pair qu'elle comprend les remplissages décrit par soi-même quand le remplissage est ajouté à l'extrémité du champ information de PPP.

Cette option est la plus susceptible d'être utilisée quand on configure quelques protocoles, tels que les protocoles de couche réseau ou de compression, qui exigent la détection et l'enlèvement de n'importe quel remplissage de fin. De tels protocoles spéciaux sont identifiés dans leurs documents respectifs.

Si l'option est rejetée, le pair NE DOIT ajouter aucun remplissage aux protocoles spéciaux identifiés, mais PEUT ajouter le remplissage à d'autres protocoles.

Si l'option est validée, le pair DOIT suivre les procédures pour ajouter les remplissages décrit par soi-même, mais seulement aux protocoles spécifiquement identifiés. Il n'est pas obligatoire que le pair ajoute un remplissage à d'autres protocoles.

Notes de Mise en place :

Ceci est défini de sorte que le rejet manipule l'un ou l'autre cas où le pair ne produit pas des remplissages décrit par soi-même. Quand le pair ne produit jamais de remplissage, il peut sans risque rejeter l'option. Quand le pair ne comprend pas l'option, il ne configurera pas également avec succès un protocole spécial qui exige l'élimination des remplissages.

Tandis que quelques expéditeurs pourraient seulement être capables d'ajouter un remplissage à chaque protocole ou de ne pas ajouter de remplissage à n'importe quel protocole, par conception le récepteur n'a pas besoin d'examiner ces protocoles qui n'ont pas besoin d'éliminer le remplissage.

Pour éviter les dialogues de configuration inutiles, une mise en place qui produit du remplissage, et qui a un protocole configuré qui exige que le remplissage soit connu, DEVRAIT inclure cette option dans sa "Demande-Configuration", et DEVRAIT "Configuration-NAK" avec cette option quand elle n'est pas présente dans la demande du pair.

Chaque octet du remplissage décrit par soi-même contient l'index de cet octet. Le premier octet de remplissage DOIT contenir la valeur un (1), qui indique le protocole de remplissage à l'option "Trames-Composées". Après avoir retiré la FCS, l'octet final de remplissage indique le nombre d'octets de remplissage à retirer. Par exemple, trois octets de remplissage contiendraient les valeurs 1, 2, 3.

La valeur maximum de remplissage (MPV "Maximum Pad Value") est également négociée. Seulement les valeurs de 1 à MPV sont utilisées. Quand aucun remplissage ne serait autrement exigé, mais l'octet final de la zone d'information de PPP contient la valeur 1 à MPV, au moins un octet de remplissage décrit par soi-même DOIT être ajouté à la trame. Si l'octet final est plus grand que MPV, aucun remplissage supplémentaire n'est exigé.

Notes de Mise en place :

Si un quelconque de ces octets de remplissage contient une valeur d'incrément incorrecte, la trame entière DEVRAIT être silencieusement jetée. Ceci afin d'empêcher la confusion avec l'option "FCS-Alternative", mais pourrait ne pas être nécessaire dans des réalisations robustes.

Puisque cette option est destinée à supporter des protocoles de compression, la valeur maximum de remplissage (MPV) est indiquée pour limiter la probabilité qu'une trame puisse réellement devenir plus longue.

Un récapitulatif du format d'option de configuration de remplissage décrit par soi-même est montré ci-dessous. Les champs sont transmis de gauche à droite.

0										1										2									
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3						
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+																													
Type										Longueur										Maximum									
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+																													

Type : 10

Longueur : 3

Maximum : Ce champ indique le plus grand nombre d'octets de remplissage qui peuvent être ajoutés à la trame. La valeur peut s'étendre de 1 à 255, mais les valeurs de 2, 4 ou 8 sont les plus probables.

2.3 Rappel

Description

Cette option de configuration fournit une méthode pour une mise en place pour demander à un pair sur réseau commuté de rappeler. Cette option pourrait être utilisée pour beaucoup de buts divers, tels que la réduction des frais de communication.

Quand le rappel de service est négocié avec succès, et que l'authentification est complète, la phase d'authentification procède directement à la phase d'arrêt, et le lien est déconnecté.

Puis, le pair rétablit le lien, sans négociation du rappel de service.

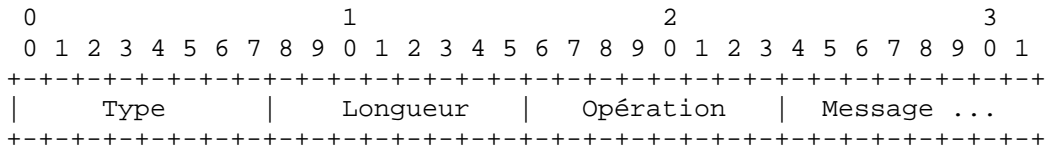
Notes de Mise en place :

Un pair qui est d'accord sur cette option DEVRAIT demander l'option de configuration du protocole d'authentification. L'information d'utilisateur apprise pendant l'authentification peut être employée pour déterminer l'emplacement de l'utilisateur, ou pour limiter un utilisateur à certains emplacements, ou pour déterminer simplement qui sera facturé pour le service.

L'authentification DEVRAIT être demandée en retour par la mise en place quand il y a un rappel, si l'authentification mutuelle est désirée.

Un récapitulatif du format d'option de rappel de service est montré ci-dessous. Les champs sont

transmis de gauche à droite.



Type : 13

Longueur : >= 3

Exécution : Le champ Opération est d'un octet et indique le contenu du champ message.

- 0 l'emplacement est déterminé par l'authentification de l'utilisateur
- 1 chaîne de caractères de numérotation, dont le format et le contenu assume la connaissance de configuration du dispositif spécifique qui effectue le rappel de service.
- 2 identificateur d'emplacement, qui peut ou peut ne pas être lisible par l'homme, à utiliser ainsi que l'information d'authentification pour une consultation de base de données pour déterminer l'emplacement du rappel de service.
- 3 nombre E.164.
- 4 nom distinctif.

Message : Le champ message est de zéro octets ou plus, et son contenu général est déterminé par le champ opération. Le format réel de l'information est spécifique au site ou à l'application, et une mise en place robuste DEVRAIT supporter le champ comme octets non distingués. La taille est déterminée à partir du champ longueur.

Il est intentionnel que seulement un utilisateur autorisé aura l'information spécifique au site correcte pour se servir du rappel de service. La codification de l'intervalle de l'utilisation permise de ce champ est en dehors de la portée de cette spécification.

2.4 Trames Composées

Description

Cette option de configuration fournit une méthode pour une mise en place pour envoyer des multiples paquets PPP encapsulés dans la même trame. Cette option pourrait être utilisée pour beaucoup de buts divers, tels que la réduction des frais.

Seulement les protocoles de PPP qui ont des longueurs déterminées ou des champs longueur

intégrale peuvent être agrégés dans une trame composée.

Quand des trames composées sont négociées avec succès, l'expéditeur PEUT ajouter des paquets supplémentaires à la même trame. Chaque paquet est immédiatement suivi d'un autre champ de protocole, avec son datagramme propre.

Quand du remplissage est ajouté à l'extrémité du champ information, le procédé décrit dans Remplissage décrit par soi-même est utilisé. Par conséquent, cette option DOIT être négociée ainsi que l'option de Remplissage décrit par soi-même.

Si l'option "FCS-Alternative" a été négociée, le remplissage décrit par soi-même DOIT toujours être ajouté. C'est-à-dire que le paquet final DOIT être suivi d'une série d'octets, le premier d'entre eux contenant la valeur un (1).

À la réception, le premier champ de protocole est examiné, et le paquet est traité comme d'habitude. Pour ces datagrammes qui ont une longueur déterminée, le reste de la trame est retourné au démultiplexeur. Chaque champ de protocole aboutissant est traité comme un paquet séparé. Ce traitement est complet quand un paquet qui n'a pas une longueur déterminée est traité, quand le reste de la trame est vide, ou quand le champ de protocole est déterminé pour avoir une valeur de un (1).

La valeur de protocole de PPP de un (1) est réservée comme protocole de remplissage. Tous les octets suivants sont retirés comme étant du remplissage.

Un récapitulatif du format d'option de trames composées est montré ci-dessous. Les champs sont transmis de gauche à droite.

```

      0                               1
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-----+-----+-----+-----+
|           Type           | Longueur |
+-----+-----+-----+-----+

```

Type : 15

Longueur : 2

2.4.1 Considérations de LCP

Pendant la négociation initiale, l'option de Trames composées peut être employée pour réduire au minimum le temps d'attente de négociation, en réduisant le nombre de trames échangées.

Les premiers paquets de demandes de configuration LCP sont envoyés comme d'habitude dans une trame simple, y compris les options de remplissage décrit par soi-même et de Trames composées.

Le pair DEVRAIT répondre avec un "Configuration-ACK", suivi dans une trame composée de sa demande de configuration LCP, et de n'importe quelle demande de configuration NCP désirée.

Sur la réception, l'implantation locale DEVRAIT traiter le "Configuration-ACK" comme d'habitude. Puisque le pair a accepté d'envoyer des trames composées, l'implantation DOIT examiner le reste de la trame pour des paquets supplémentaires. Si le pair indiquait également les options remplissage décrit par soi-même et des Trames composées dans sa demande de configuration, l'implantation locale DEVRAIT maintenir son "Configuration-ACK", et d'autres paquets de configuration de NCP DEVRAIENT être ajoutés à la trame de retour.

En même temps que la trame de retour finale du pair, le nombre minimum de trames pour terminer la configuration est 4.

A. Implantation Rapide du Contrôle de Trame (FCS)

A.1. Méthode de Calcul du FCS 32 bits

Le code suivant fournit une table de computation pour calculer le contrôle de trame 32 bits pendant que les données arrivent à l'interface.

```
* u32 représente un nombre 32-bit non signé. Ajuster le "typedef" selon
* votre matériel.
*/
typedef unsigned long u32;
```

```
static u32 fcstab_32[256] =
{
    0x00000000, 0x77073096, 0xee0e612c, 0x990951ba,
    0x076dc419, 0x706af48f, 0xe963a535, 0x9e6495a3,
    0x0edb8832, 0x79dcb8a4, 0xe0d5e91e, 0x97d2d988,
    0x09b64c2b, 0x7eb17cbd, 0xe7b82d07, 0x90bf1d91,
    0x1db71064, 0x6ab020f2, 0xf3b97148, 0x84be41de,
    0x1adad47d, 0x6ddde4eb, 0xf4d4b551, 0x83d385c7,
    0x136c9856, 0x646ba8c0, 0xfd62f97a, 0x8a65c9ec,
    0x14015c4f, 0x63066cd9, 0xfa0f3d63, 0x8d080df5,
    0x3b6e20c8, 0x4c69105e, 0xd56041e4, 0xa2677172,
    0x3c03e4d1, 0x4b04d447, 0xd20d85fd, 0xa50ab56b,
    0x35b5a8fa, 0x42b2986c, 0xdbbbc9d6, 0xacbcf940,
    0x32d86ce3, 0x45df5c75, 0xdcd60dcf, 0xabd13d59,
    0x26d930ac, 0x51de003a, 0xc8d75180, 0xbf061116,
    0x21b4f4b5, 0x56b3c423, 0xcfba9599, 0xb8bda50f,
    0x2802b89e, 0x5f058808, 0xc60cd9b2, 0xb10be924,
    0x2f6f7c87, 0x58684c11, 0xc1611dab, 0xb6662d3d,
    0x76dc4190, 0x01db7106, 0x98d220bc, 0xefd5102a,
    0x71b18589, 0x06b6b51f, 0x9fbfe4a5, 0xe8b8d433,
    0x7807c9a2, 0x0f00f934, 0x9609a88e, 0xe10e9818,
    0x7f6a0dbb, 0x086d3d2d, 0x91646c97, 0xe6635c01,
    0x6b6b51f4, 0x1c6c6162, 0x856530d8, 0xf262004e,
    0x6c0695ed, 0x1b01a57b, 0x8208f4c1, 0xf50fc457,
    0x65b0d9c6, 0x12b7e950, 0x8bbeb8ea, 0xfcb9887c,
    0x62dd1ddf, 0x15da2d49, 0x8cd37cf3, 0xfbd44c65,
```

```

0x4db26158, 0x3ab551ce, 0xa3bc0074, 0xd4bb30e2,
0x4adfa541, 0x3dd895d7, 0xa4d1c46d, 0xd3d6f4fb,
0x4369e96a, 0x346ed9fc, 0xad678846, 0xda60b8d0,
0x44042d73, 0x33031de5, 0xaa0a4c5f, 0xdd0d7cc9,
0x5005713c, 0x270241aa, 0xbe0b1010, 0xc90c2086,
0x5768b525, 0x206f85b3, 0xb966d409, 0xce61e49f,
0x5edef90e, 0x29d9c998, 0xb0d09822, 0xc7d7a8b4,
0x59b33d17, 0x2eb40d81, 0xb7bd5c3b, 0xc0ba6cad,
0xedb88320, 0x9abfb3b6, 0x03b6e20c, 0x74b1d29a,
0xead54739, 0x9dd277af, 0x04db2615, 0x73dc1683,
0xe3630b12, 0x94643b84, 0x0d6d6a3e, 0x7a6a5aa8,
0xe40ecf0b, 0x9309ff9d, 0x0a00ae27, 0x7d079eb1,
0xf00f9344, 0x8708a3d2, 0x1e01f268, 0x6906c2fe,
0xf762575d, 0x806567cb, 0x196c3671, 0x6e6b06e7,
0xfed41b76, 0x89d32be0, 0x10da7a5a, 0x67dd4acc,
0xf9b9df6f, 0x8ebeeff9, 0x17b7be43, 0x60b08ed5,
0xd6d6a3e8, 0xa1d1937e, 0x38d8c2c4, 0x4fdff252,
0xd1bb67f1, 0xa6bc5767, 0x3fb506dd, 0x48b2364b,
0xd80d2bda, 0xaf0alb4c, 0x36034af6, 0x41047a60,
0xdf60efc3, 0xa867df55, 0x316e8eef, 0x4669be79,
0xcb61b38c, 0xbc66831a, 0x256fd2a0, 0x5268e236,
0xcc0c7795, 0xbb0b4703, 0x220216b9, 0x5505262f,
0xc5ba3bbe, 0xb2bd0b28, 0x2bb45a92, 0x5cb36a04,
0xc2d7ffa7, 0xb5d0cf31, 0x2cd99e8b, 0x5bdeae1d,
0x9b64c2b0, 0xec63f226, 0x756aa39c, 0x026d930a,
0x9c0906a9, 0xeb0e363f, 0x72076785, 0x05005713,
0x95bf4a82, 0xe2b87a14, 0x7bb12bae, 0x0cb61b38,
0x92d28e9b, 0xe5d5be0d, 0x7cdcefb7, 0x0bdbdf21,
0x86d3d2d4, 0xf1d4e242, 0x68ddb3f8, 0x1fda836e,
0x81be16cd, 0xf6b9265b, 0x6fb077e1, 0x18b74777,
0x88085ae6, 0xff0f6a70, 0x66063bca, 0x11010b5c,
0x8f659eff, 0xf862ae69, 0x616bffd3, 0x166ccf45,
0xa00ae278, 0xd70dd2ee, 0x4e048354, 0x3903b3c2,
0xa7672661, 0xd06016f7, 0x4969474d, 0x3e6e77db,
0xaed16a4a, 0xd9d65adc, 0x40df0b66, 0x37d83bf0,
0xa9bcae53, 0xdebb9ec5, 0x47b2cf7f, 0x30b5ffe9,
0xbdbdf21c, 0xcabac28a, 0x53b39330, 0x24b4a3a6,
0xbad03605, 0xcdd70693, 0x54de5729, 0x23d967bf,
0xb3667a2e, 0xc4614ab8, 0x5d681b02, 0x2a6f2b94,
0xb40bbe37, 0xc30c8ea1, 0x5a05df1b, 0x2d02ef8d
};

```

```

#define PPPINITFCS32 0xffffffff /* valeur Initiale FCS */
#define PPPGOODFCS32 0xdebb20e3 /* bonne valeur finale FCS */

/*
 * Calculer un nouveau FCS en donnant le FCS courant et la nouvelle donnée.
 */
u32 pppfcs32(fcs, cp, len)
    register u32 fcs;
    register unsigned char *cp;
    register int len;
    {
    ASSERT(sizeof (u32) == 4);
    ASSERT(((u32) -1) > 0);
    while (len--)
        fcs = (((fcs) >> 8) ^ fcstab_32[((fcs) ^ (*cp++)) & 0xff]);
    }

```

```

    return (fcs);
}

/*
 * Comment utiliser le fcs
 */
tryfcs32(cp, len)
    register unsigned char *cp;
    register int len;
{
    u32 trialfcs;

    /* ajout sur sortie */
    trialfcs = pppfcs32( PPPINITFCS32, cp, len );
    trialfcs ^= 0xffffffff;          /* complément */
    cp[len] = (trialfcs & 0x00ff);   /* octet moins significatif en
premier */
    cp[len+1] = ((trialfcs >>= 8) & 0x00ff);
    cp[len+2] = ((trialfcs >>= 8) & 0x00ff);
    cp[len+3] = ((trialfcs >> 8) & 0x00ff);

    /* saisi sur entrée */
    trialfcs = pppfcs32( PPPINITFCS32, cp, len + 4 );
    if ( trialfcs == PPPGOODFCS32 )
        printf("Good FCS\n");
}

```

Considérations Sécuritaires

Les problèmes de sécurité sont brièvement discutés dans les sections concernant l'option de configuration du rappel de service.

Références

- [1] Simpson, W., Editor, "The Point-to-Point Protocol (PPP)", RFC 1548, Daydreamer, December 1993.
- [2] Reynolds, J., and J. Postel, "Assigned Numbers", STD 2, RFC 1340, USC/Information Sciences Institute, July 1992.
- [3] Simpson, W., Editor, "PPP in HDLC Framing", RFC 1549, Daydreamer, December 1993.

Remerciements

Le dispositif d'identification a été suggéré par Bob Sutterfield (Morning Star Technologies).

Le dispositif Temps-Restant a été suggéré par Brad Parker (FCR).

Une partie du texte initial pour FCS-Alternatives a été fournie par Arthur Harvey (DEC). La FCS nulle a été demandée par Peter Honeyman (UMich). L'exemple de code de FCS 32 bits a été fourni par Karl Fox (Morning Star Technologies).

Le remplissage décrit par soi-même a été suggérée et nommé par Fred Baker (ACC).

Les Trames composées ont été suggérées par Keith Sklower (Berkeley).

Remerciements spéciaux à "Morning Star Technologies" pour la fourniture du support de ressources informatiques et d'accès au réseau pour écrire cette spécification.

Adresse du comité

Le groupe de travail peut être contacté via le siège courant :

Fred Baker
Advanced Computer Communications
315 Bollay Drive
Santa Barbara, California 93117

E-Mail: fbaker@acc.com

Adresse de l'éditeur

Les questions sur ce document peuvent aussi être adressée à :

William Allen Simpson
Daydreamer
Computer Systems Consulting Services
1384 Fontaine
Madison Heights, Michigan 48071

E-Mail: Bill.Simpson@um.cc.umich.edu
bsimpson@MorningStar.com