

Groupe de travail Réseau
Request for Comments : 4873
 RFC mise à jour : 3473, 4872
 Catégorie : Sur la voie de la normalisation
 Traduction Claude Brière de L'Isle

L. Berger, LabN Consulting
 I. Bryskin, ADVA Optical
 D. Papadimitriou, Alcatel
 A. Farrel, Old Dog Consulting
 mai 2007

Récupération de segment GMPLS

Statut du présent mémoire

Le présent document spécifie un protocole de l'Internet en cours de normalisation 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 "Protocoles officiels de l'Internet" (STD 1) pour voir l'état de normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

(La présente traduction incorpore les errata 937, 935, 939, 943, 1797.)

Notice de Copyright

Copyright (C) The IETF Trust (2007).

Résumé

Le présent document décrit les procédures spécifiques de protocole pour les extensions de signalisation de l'ingénierie du trafic du protocole de réservation de ressources (RSVP-TE, *Resource ReserVation Protocol - Traffic Engineering*) de la commutation d'étiquettes multi protocoles généralisée (GMPLS, *Generalized Multi-Protocol Label Switching*) pour la prise en charge de la protection et restauration de segment de chemin de commutation d'étiquettes (LSP, *Label Switched Path*). Ces extensions sont destinées à compléter et être en cohérence avec les extensions à RSVP-TE pour la récupération de bout en bout de GMPLS (RFC4872). Les implications et interactions avec le réacheminement rapide (*fast reroute*) sont aussi traitées. Le présent document met aussi à jour le traitement des objets NOTIFY_REQUEST.

Table des matières

1. Introduction.....	2
1.1 Conventions utilisées dans ce document.....	2
2. Récupération de segment.....	2
2.1 Protection de segment.....	3
2.2 Réacheminement et restauration de segment	4
3. Objet ASSOCIATION.....	4
3.1 Format.....	4
3.2 Procédures.....	4
4. Contrôle explicite de récupération de segment de LSP.....	5
4.1 Format d'objet Chemin secondaire explicite.....	5
4.2 Procédures de contrôle explicite.....	5
4.3 Suppression à partir de nœuds non d'entrée.....	7
5. Objets Chemin d'enregistrement secondaire.....	8
5.1 Format.....	8
5.2 Traitement de Path.....	8
5.3 Traitement de Resv.....	9
6. Contrôle dynamique de récupération de segment de LSP.....	9
6.1 Format d'objet PROTECTION modifié.....	9
6.2 Procédures de contrôle dynamique.....	10
7. Formats de message RSVP mis à jour.....	10
8. Considérations sur la sécurité.....	12
9. Considérations relatives à l'IANA.....	12
9.1 Allocation d'un nouveau type d'association.....	12
9.2 Définition des bits réservés d'objet PROTECTION.....	12
9.3 Objet Chemin secondaire explicite.....	12
9.4 Objet Chemin d'enregistrement secondaire.....	13
9.5 Nouveau code d'erreur.....	13
9.6 Utilisation du C-type d'objet PROTECTION.....	13
10. Références.....	13

10.1 Références normatives.....	13
10.2 Références pour information.....	13
Adresse des auteurs.....	14
Déclaration complète de droits de reproduction.....	14

1. Introduction

La [RFC4427] couvre plusieurs types de protection, incluant des approches de bout en bout et fondées sur le segment. Les "Extensions à RSVP-TE pour la prise en charge de la récupération de commutation d'étiquettes multi protocoles généralisée (GMPLS) de bout en bout" [RFC4872] définissent un ensemble d'extensions pour prendre en charge plusieurs types de récupération. Les types pris en charge incluent la protection 1+1 unidirectionnelle/bidirectionnelle, la protection de LSP avec trafic supplémentaire (incluant la protection 1:N avec trafic supplémentaire), le réacheminement pré-programmé de LSP sans trafic supplémentaire (incluant le maillage partagé) et le réacheminement complet de LSP. Dans tous les cas, la récupération est fournie de bout en bout, c'est-à-dire, que les nœuds d'entrée et de sortie des deux LSP, le protégé et le protecteur, sont les mêmes.

La [RFC4090] fournit une forme de récupération de segment pour les réseaux de paquets MPLS-TE. Deux méthodes de réacheminement rapide sont définies dans la [RFC4090]. La méthode de sauvegarde de un à un crée des LSP de détournement pour chaque LSP protégé à chaque point potentiel de réparation local. La méthode de sauvegarde de facilité crée un tunnel de contournement pour protéger un potentiel point de défaillance qui est partagé par plusieurs LSP et utilise l'empilage d'étiquette. Aucune de ces deux approches ne prend en charge l'ensemble complet de types de récupération pris en charge par la [RFC4872]. De plus, la méthode de sauvegarde de facilité n'est pas applicable à la plupart des technologies de commutation qui ne sont pas de paquets.

Les extensions définies dans le présent document permettent la prise en charge de l'ensemble complet de types de récupération pris en charge par la [RFC4872], mais sur la base d'un segment, ou portion du LSP. Les extensions permettent (a) de signaler le type désiré de protection de segment de LSP, (b) aux nœuds en amont d'identifier facultativement où commence et où finit la protection de segment, (c) l'identification facultative des bonds utilisés sur les segments de protection, et (d) le rapport des chemins utilisés pour protéger un LSP. Les extensions élargissent aussi la portée topologique sur laquelle la protection peut être prise en charge. Elles permettent des segments de récupération qui protègent contre un nombre arbitraire de nœuds et liaisons. Elles permettent des protections qui se chevauchent et des protections incorporées. Ces extensions sont destinées à être compatibles avec le réacheminement rapide, et dans certains cas à être utilisées avec le réacheminement rapide.

1.1 Conventions utilisées dans ce document

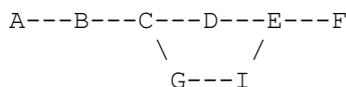
Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" en majuscules dans ce document sont à interpréter comme décrit dans le BCP 14, [RFC2119].

De plus, le lecteur est supposé être familier de la terminologie utilisée dans les [RFC3209], [RFC3471], et [RFC3473], ainsi que les [RFC4427], [RFC4426], [RFC4872], et [RFC4090].

2. Récupération de segment

La récupération de segment est utilisée pour fournir la protection et la restauration sur une portion d'un LSP de bout en bout. Une telle protection et restauration de segment est utile pour protéger contre une défaillance de portée, une défaillance de nœud, ou une défaillance sur une portion particulière d'un réseau utilisé par un LSP.

Considérons la topologie suivante :



Dans cette topologie, la protection et récupération de bout en bout n'est pas possible pour un LSP allant du nœud A au nœud F, mais il est possible de protéger/récupérer une portion du LSP. Précisément, si le LSP utilise un chemin actif de

[A,B,C,D,E,F], alors un LSP de protection ou restauration peut être établi le long du chemin [C,G,I,E]. Ce LSP protège contre les défaillances sur les portées {C,D} et {D,E}, ainsi que d'une défaillance du nœud D. Cette forme de protection/restauration est appelée la protection de segment et la restauration de segment, ou collectivement récupération de segment. Le LSP qui fournit la protection ou restauration est appelé un LSP de protection de segment ou un LSP de restauration de segment. Le terme de "LSP de récupération de segment" est utilisé pour couvrir un LSP de protection de segment ou un LSP de restauration de segment. Le terme de "nœud de branche" est utilisé pour se référer à un nœud qui initie un LSP de récupération, par exemple, le nœud C dans la figure ci-dessus. Ceci est équivalent au point de réparation local (PLR, *Point of Local Repair*) utilisé dans la [RFC4090]. Comme avec la [RFC4090], le terme de "nœud de fusion" est utilisé pour se référer à un nœud qui termine un LSP de récupération, par exemple, le nœud E dans la figure ci-dessus.

La protection ou restauration de segment est signalée en utilisant un LSP actif et un ou plusieurs LSP de récupération de segment. Chaque LSP de récupération de segment est signalé comme un LSP indépendant. Précisément, l'objet `Sender_Template` utilise l'adresse IP du nœud à l'origine du chemin de récupération, par exemple, le nœud C dans la topologie montrée ci-dessus, et l'objet `Session` contient l'adresse IP du nœud qui termine le chemin de récupération, par exemple, le nœud E montré ci-dessus. Il n'y a pas d'exigence spécifique de valeur d'identifiant de LSP, d'identifiant de tunnel, ni d'identifiant étendu de tunnel. Les valeurs de ces champs sont choisies normalement, incluant la considération du concept de "faire avant de couper" décrit dans la [RFC3209]. Les nœuds intermédiaires suivent les procédures standard de signalisation lors du traitement des LSP de récupération de segments. Un LSP de récupération de segment peut être lui-même protégé en utilisant la protection/restauration de segment ou de bout en bout. Noter que dans les environnements de PSC, il peut être souhaitable de construire les objets `Sender_Template` et `Session` selon la [RFC4090].

Quand la [RFC4090] n'est pas utilisée, l'association entre les LSP de récupération de segment avec d'autres LSP est indiquée en utilisant l'objet `ASSOCIATION` défini dans la [RFC4872]. L'objet `ASSOCIATION` est utilisé pour associer les LSP de récupération avec le LSP qu'ils protègent. Les LSP actifs et de protection, ainsi que les LSP primaires et secondaires, sont identifiés en utilisant l'état de LSP comme décrit dans la [RFC4872]. Le bit O dans la portion fanions de segment de l'objet `PROTECTION` est utilisé pour identifier quand un LSP de récupération porte le trafic normal (actif).

Un nœud en amont peut permettre aux nœuds en aval d'identifier de façon dynamique les points de branche et de fusion en réglant les bits de protection de segment de LSP désirés dans l'objet `PROTECTION`. Ces bits sont définis ci-dessous.

Facultativement, un nœud en amont, généralement le nœud d'entrée, peut identifier les points d'extrémité d'un LSP de récupération de segment. Ceci est accompli en utilisant un nouvel objet. Cet objet utilise le même format qu'un objet `Chemin explicite (ERO, Explicit Route Object)` et est appelé un objet de chemin explicite secondaire (`SERO, Secondary Explicit Route object`) ; voir au paragraphe 4.1. Les `SERO` prennent aussi en charge un nouveau sous objet pour indiquer le type de protection ou de restauration à fournir. Au minimum, un `SERO` va indiquer l'initiateur d'un LSP de récupération, un type de protection/restauration et une terminaison. La sémantique standard de l'`ERO` [RFC3209] peut facultativement être utilisée au sein du `SERO` pour contrôler explicitement le LSP de récupération. Un objet `Chemin d'enregistrement secondaire (SRRO, Secondary Record Route Object)` est défini pour enregistrer le chemin d'un LSP de récupération de segment ; voir la Section 5.

Les `SERO` sont portés entre le nœud qui crée le `SERO`, normalement l'entrée, et le nœud qui initie un LSP de récupération. Le nœud initiateur d'un LSP de récupération utilise le `SERO` pour créer le `ERO` pour le LSP de récupération. À ce nœud (branche) tous les objets locaux sont supprimés, et le nouveau sous objet de protection est utilisé pour créer l'objet `PROTECTION` pour le LSP de récupération. Il est aussi possible de contrôler le traitement d'une défaillance pour établir un LSP de récupération.

Les `SRRO` sont portés dans des messages `Path` entre le nœud qui termine un LSP de récupération, le nœud de fusion, et la sortie. Les `SRRO` sont utilisés dans les messages entre un nœud de branche et l'entrée. Le nœud de fusion d'un LSP de récupération crée un `SRRO` en copiant le `RRO` du message `Path` du LSP de récupération associé dans un nouvel objet `SRRO`. Tous les `SRRO` présents dans le message `Path` du LSP de récupération sont aussi copiés. Le nœud de branche d'un LSP de récupération crée un `SRRO` en copiant le `RRO` provenant du message `Resv` du LSP de récupération associé dans un nouvel objet `SRRO`. Tout `SRRO` présent dans le message `Resv` du LSP de récupération est aussi copié.

Le traitement des demandes `Notify` est aussi impacté par la récupération de segment de LSP. Selon la [RFC3473], seulement un objet `NOTIFY_REQUEST` est significatif et devrait être propagé. Les objets `NOTIFY_REQUEST` supplémentaires sont utilisés pour identifier les nœuds de branche de LSP de récupération.

2.1 Protection de segment

Trois approches pour la protection de bout en bout sont définies dans la [RFC4872] : protection 1+1 unidirectionnelle

(Section 5) protection 1+1 bidirectionnelle (Section 6) et protection 1:1 avec trafic supplémentaire (Section 7). Les formes de protection de segment de ces approches de protection fonctionnent toutes comme leur contrepartie de bout en bout. Chacune se comporte juste comme sa contrepartie de bout en bout, avec l'exception que le LSP de protection protège seulement une portion du LSP actif. Le type de protection à utiliser sur un LSP de protection de segment est indiqué, à l'entrée du LSO de protection, en utilisant le sous objet de protection SERO défini au paragraphe 4.1.

Le traitement de la commutation pour la protection de segment 1+1 bidirectionnelle et la protection 1:1 avec trafic supplémentaire suit les mêmes procédures que les formes de protection de bout en bout ; voir les détails aux paragraphes 6.2 et 7.2 de la [RFC4872].

2.2 Réacheminement et restauration de segment

Trois approches de réacheminement et de restauration sont définies dans la [RFC4872] : réacheminement sans trafic supplémentaire (Section 8) restauration de maillage partagé (Section 9) réacheminement de LSP (complet) (Section 11). Comme avec la protection, ces approches sont prises en charge sur la base du segment. Les formes de segment du réacheminement et de la restauration opèrent exactement comme leurs contreparties de bout en bout, à l'exception que le LSP de restauration recouvre seulement une portion du LSP actif. Le type de réacheminement ou de restauration à utiliser sur un LSP de restauration de segment est indiqué, à l'entrée du LSP de restauration, en utilisant le nouveau sous objet de protection SERO.

3. Objet ASSOCIATION

L'objet ASSOCIATION est utilisé pour l'association des LSP de protection de segment quand la [RFC4090] n'est pas utilisée. L'objet ASSOCIATION est défini dans la [RFC4872]. Dans le présent document, on définit une nouvelle valeur de champ Type d'association pour prendre en charge "faire avant de couper" ; les formats et procédures définis dans la [RFC4872] ne sont par ailleurs pas modifiés.

3.1 Format

Type d'association : 16 bits

Valeur Type

2 Partage de ressource (R)

Voir dans la [RFC4872] la définition des autres champs et valeurs.

3.2 Procédures

L'objet ASSOCIATION est utilisé pour associer les différents LSP les uns avec chaque autre. Dans le contexte de protection et restauration, l'objet est utilisé pour associer un LSP de récupération avec le LSP qu'il protège. L'objet ASSOCIATION est aussi utilisé pour prendre en charge le partage de ressources durant un "faire avant de couper". Cet objet NE DOIT PAS être utilisé quand une association est faite selon les méthodes définies dans la [RFC4090].

3.2.1 Traitement du type de récupération

Les procédures de traitement du type de récupération sont les mêmes que celles définies dans la [RFC4872], mais le traitement et l'identification se produisent par rapport au LSP de récupération de segment. Noter que cela signifie que plusieurs objets ASSOCIATION de type récupération peuvent être présents sur un LSP.

3.2.2 Traitement du type d'association de partage de ressource

L'objet ASSOCIATION avec un type d'association de valeur Partage de ressource est utilisé pour activer le partage de ressources durant le faire avant de couper. Le partage de ressource durant le faire avant de couper est défini dans la [RFC3209]. La prise en charge définie ne fonctionne qu'avec les LSP qui partagent la même sortie de LSP. Avec l'introduction du LSP de récupération de segment, il est maintenant possible à un point d'extrémité de LSP de changer durant le faire avant de couper.

Un nœud inclut un objet ASSOCIATION avec un type d'association de ressources dans un message Path sortant quand il souhaite indiquer le partage de ressources à travers un ensemble de LSP associés. La source d'association est réglée à l'adresse de routeur du nœud d'origine. L'identifiant d'association DOIT être réglé à une valeur qui identifie de façon univoque l'association des LSP. Ce PEUT être réglé à l'identifiant de LSP du LSP actif. Une fois inclus, un objet ASSOCIATION avec un type d'association de partage de ressource NE DEVRAIT PAS être supprimé des messages Path associés à un LSP.

Tout nœud qui traite un message Path pour lequel le nœud n'a pas d'état correspondant, et qui contient un objet ASSOCIATION avec un type de partage de ressource, examine les LSP existants avec des valeurs de type d'association, source d'association, et identifiant d'association correspondantes. Si une correspondance est trouvée, alors le partage de ressource de style [RFC3209] DEVRAIT être fourni entre les LSP, nouveau et ancien. Voir des détails supplémentaires dans la [RFC3209].

4. Contrôle explicite de récupération de segment de LSP

Les objets Chemin secondaire explicite, ou SERO, sont définis dans le présent document. Ils peuvent être utilisés pour indiquer les nœuds de branche et de fusion des LSP de récupération. Ils peuvent aussi fournir des informations supplémentaires qui sont à porter dans l'ERO d'un LSP de récupération. Quand le contrôle en amont des nœuds de branche et de fusion n'est pas désiré, les SERO ne sont pas utilisés.

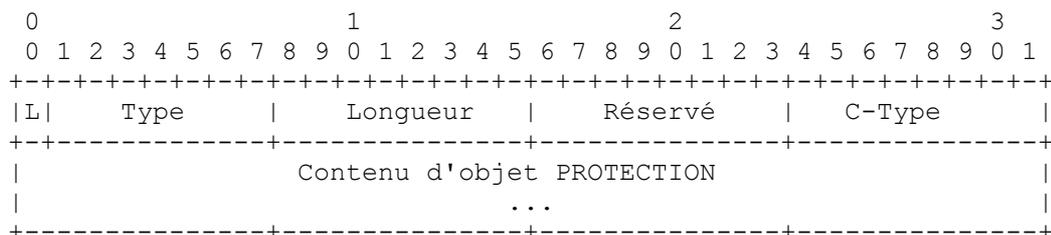
4.1 Format d'objet Chemin secondaire explicite

Le format d'un objet SECONDARY_EXPLICIT_ROUTE est le même que celui de l'objet EXPLICIT_ROUTE. Cela inclut la définition de sous objets définis pour l'objet EXPLICIT_ROUTE. La classe de l'objet SECONDARY_EXPLICIT_ROUTE est 200 (de forme 11bbbbbb).

4.1.1 Sous objet Protection

Un nouveau sous objet, appelé sous objet Protection, est défini pour être utilisé dans l'objet SECONDARY_EXPLICIT_ROUTE. Comme mentionné ci-dessus, le nouveau sous objet Protection est utilisé pour créer l'objet PROTECTION pour le LSP de récupération. Des procédures spécifiques relatives au sous objet Protection sont fournies au paragraphe 4.2. Le sous objet Protection n'est pas valide utilisé avec les objets Explicit et Record Route et NE DOIT PAS être inclus dans ces objets.

Le format du sous objet Protection est défini comme suit :



L : ce bit est défini dans la [RFC3209] et DOIT être réglé à zéro pour les sous objets Protection.

Type : 37 Protection

Longueur : comme défini au paragraphe 4.3.3 de la [RFC3209].

Réserve : ce champ est réservé. Il DOIT être réglé à zéro à l'émission et DOIT être ignoré à réception.

C-Type : C-Type de l'objet PROTECTION inclus.

Contenu d'objet PROTECTION : le contenu de l'objet PROTECTION, avec le format correspondant au C-Type indiqué, en excluant l'en-tête d'objet.

4.2 Procédures de contrôle explicite

Les SERO sont portés dans les messages Path et indiquent à quel nœud un LSP de récupération va être initié par rapport au LSP portant le SERO. Plus d'un SERO PEUT être présent dans un message Path.

Pour indiquer les nœuds de branche et de fusion d'un LSP de récupération, un SERO est créé et ajouté au message Path du LSP à récupérer. La décision de créer et insérer un SERO est une affaire locale et sort du domaine d'application du présent document.

Un SERO DEVRAIT contenir au moins trois sous objets. Le premier sous objet DOIT indiquer le nœud qui va être l'origine du LSP de récupération, c'est-à-dire le nœud de branche de segment. L'adresse utilisée DEVRAIT aussi être mentionnée dans l'ERO ou un autre SERO. Cela assure que le nœud de branche est le long du chemin du LSP. Le second sous objet DEVRAIT être un sous objet Protection et devrait indiquer la protection ou restauration à fournir par le LSP de récupération. Quand le sous objet Protection est présent, les fanions de récupération de segment de LSP dans le sous objet Protection DOIVENT être ignorés. Le sous objet final dans le SERO DOIT être le nœud de fusion du LSP de récupération, et PEUT avoir le bit L établi. Des sous objets ERO standard PEUVENT être insérés entre le sous objet Protection et le sous objet final. Ces sous objets PEUVENT être lâches ou stricts.

Un nœud qui reçoit un message Path contenant un ou plusieurs SERO DEVRAIT examiner chaque SERO pour voir si il indique un point de branche locale. Cette détermination est faite en examinant le premier objet de chaque SERO et voir si l'adresse indiquée dans le sous objet peut être associée au nœud local. Si une des adresses indiquée est associée au nœud local, alors le nœud local est un nœud de branche. Si le nœud local n'est pas un nœud de branche, tous les SERO reçus DOIVENT être transmis, sans modification, dans le message Path sortant correspondant.

À un nœud de branche, le SERO, avec le message Path du LSP à récupérer, fournit les informations pour créer le LSP de récupération. Le message Path pour le LSP de récupération est créé au nœud de branche en répliquant les objets portés dans le message Path entrant du LSP protégé. Certains objets sont remplacés ou modifiés dans le message Path sortant du LSP de récupération. L'objet `Sender_template` DOIT être mis à jour pour utiliser une adresse (dans son champ Adresse d'envoyeur du tunnel) sur le nœud local, et l'identifiant de LSP DOIT être mis à jour pour assurer son unicité. L'objet `Session` DOIT être mis à jour pour utiliser l'adresse indiquée dans le sous objet final du SERO comme adresse de point d'extrémité de tunnel, l'identifiant de tunnel PEUT être mis à jour, et l'identifiant étendu de tunnel DOIT être réglé à l'adresse du nœud local. L'objet `PROTECTION` est remplacé par le contenu du sous objet Protection du SERO correspondant, quand il est présent. Dans tous les cas, le bit R d'un nouvel objet `PROTECTION` est remis à (0). Aucun RRO et ERO présent dans le message Path NE DOIT être inclus dans le LSP de récupération. Un nouvel ERO DOIT être inclus, avec le contenu du SERO qui indiquait une branche locale. Comme avec tous les ERO, aucune information locale (adresse locale et sous objets de protection) n'est portée dans l'ERO porté dans le message Path sortant du LSP de récupération. Le SERO qui indiquait une branche locale DOIT être omis du message Path sortant du LSP de récupération. Noter que par défaut, tous les autres SERO reçus sont passés dans le message Path sortant du LSP de récupération. Des SERO PEUVENT être omis du message Path sortant du LSP de récupération ainsi que du message Path sortant pour le LSP protégé, quand le SERO ne se rapporte pas au message Path sortant.

Le message Path résultant est utilisé pour créer le LSP de récupération. À partir de ce point, le traitement standard de message Path est utilisé dans le traitement du message Path résultant.

4.2.1 Traitement de défaillance de branche

Durant l'établissement, il est possible qu'un nœud intervenant soit incapable de prendre en charge une branche demandée. De plus, durant l'établissement et le fonctionnement normal, des messages PathErr peuvent être reçus à un nœud de branche. Le traitement de ces événements dépend d'un certain nombre de facteurs.

Quand une défaillance ou un message PathErr reçu est associé au LSP protégé, l'événement est d'abord traité selon les règles de traitement standard. Cela inclut la génération d'un message PathErr standard. Quand l'état de LSP est supprimé à cause d'une défaillance locale ou d'un message PathErr avec le fanion `Path_State_Removed` établi à (1), le nœud DOIT envoyer un message PathTear vers l'aval sur toutes les autres branches.

Quand une défaillance ou un message PathErr reçu est associé à un LSP de récupération, le traitement se fonde sur le bit R en plus du fanion `Path_State_Removed`. Dans tous les cas, un message PathErr reçu est d'abord traité selon les règles de traitement standard et la défaillance ou le message PathErr reçu DEVRAIT déclencher la génération d'un message PathErr

vers l'amont pour le LSP qui est protégé. Le message PathErr sortant DEVRAIT indiquer une erreur de "Problème d'acheminement/échec de protection du segment de LSP". Le message PathErr sortant DOIT inclure tous les SERO portés dans un message PathErr reçu. Si aucun SERO n'est présent dans un message PathErr reçu ou quand la défaillance est locale, alors un SERO qui correspond au LSP erroné ou à la branche défaillante DOIT être ajouté au message PathErr sortant.

Quand un message PathErr avec le fanion Path_State_Removed à zéro (0) est reçu, le message PathErr sortant (vers l'amont) DEVRAIT être envoyé avec le fanion Path_State_Removed à zéro (0).

Quand un message PathErr pour un LSP de récupération avec le fanion Path_State_Removed établi à (1) est reçu, le nœud qui le traite DOIT examiner le bit R (comme défini ci-dessous) du LSP qui est protégé. Le bit R est porté dans l'objet PROTECTION qui a déclenché l'initiation du LSP de récupération. Quand le bit R n'est pas établi (0), le message PathErr sortant (vers l'amont) DEVRAIT être envoyé avec le fanion Path_State_Removed à zéro (0). Quand le bit R est établi (1), le message PathErr sortant (vers l'amont) DOIT être envoyé avec le fanion Path_State_Removed établi (1).

Dans tous les cas où un message PathErr sortant (vers l'amont) est envoyé avec le fanion Path_State_Removed établi (1), tout état de chemin pour le LSP qui est protégé DOIT être supprimé, et le nœud DOIT envoyer un message PathTear vers l'aval sur toutes les branches actives.

4.2.2 Traitement du message Resv

Les nœuds de branche vont traiter les messages Resv pour les LSP de récupération et les LSP protégés. Les messages Resv sont propagés vers l'amont des nœuds de branche seulement après qu'un message Resv est reçu pour le LSP protégé. Les messages Resv sur les LSP de récupération ne vont normalement pas déclencher la transmission de messages Resv en amont (pour le LSP protégé). Des exceptions à cela incluent quand des RRO/SRRO sont collectés et durant certains traitements d'objet ADMIN_STATUS. Voir ci-dessous plus d'informations sur le traitement en cause.

4.2.3 Changement d'état administratif

En général, les objets dans un LSP de récupération sont créés sur la base des objets correspondants dans le LSP protégé. L'objet ADMIN_STATUS est créé de la même façon, mais il exige aussi une coordination particulière aux nœuds de branche. Précisément, en plus du traitement normal, un nœud de branche qui reçoit un objet ADMIN_STATUS dans un message Path DOIT aussi relayer l'objet ADMIN_STATUS dans un message Path sur chaque LSP de récupération. Tous les messages Path PEUVENT être concurremment envoyés vers l'aval.

Les nœuds en aval traitent le changement dans l'objet ADMIN_STATUS selon la [RFC3473], incluant la génération de messages Resv. Quand l'objet ADMIN_STATUS le plus récemment reçu de l'amont a le bit R établi, les nœuds de branche attendent qu'un message Resv avec un objet ADMIN_STATUS correspondant soit reçu sur toutes les branches avant de relayer un message Resv correspondant en amont.

4.2.4 Suppression de LSP de récupération

La suppression de LSP de récupération suit les procédures standard définies dans les [RFC3209] et [RFC3473]. Ces procédures incluent la suppression de LSP avec et sans établir les fanions d'état administratif décrits à la Section 7 de la [RFC3473].

4.2.4.1 Suppression sans changement d'état administratif

Le nœud qui initie la suppression génère un message PathTear. Chaque nœud qui reçoit un message PathTear traite le message PathTear selon le traitement standard (voir les [RFC3209] et [RFC2205]) et DOIT aussi relayer un message PathTear sur chaque LSP de récupération. Tous les messages PathTear (reçus de l'amont et générés en local) peuvent être envoyés concurremment vers l'aval.

4.2.4.2 Suppression avec changement d'état administratif

Selon la [RFC3473], le nœud d'entrée génère un message Path avec les bits D et R établis dans l'objet ADMIN_STATUS. La procédure de changement de l'état administratif définie au paragraphe 4.2.3 DOIT alors être suivie. Une fois que l'entrée a reçu tous les messages Resv attendus, elle DOIT suivre la procédure de suppression décrite au paragraphe 4.2.4.1.

4.3 Suppression à partir de nœuds non d'entrée

Comme avec tout LSP, chaque nœud le long d'un LSP de récupération peut initier la suppression du LSP de récupération. Pour ce faire, le nœud qui initie la suppression envoie un message PathErr avec le code d'erreur approprié et le fanion Path_State_Removed mis à zéro (0) vers le LSP d'entrée. Comme décrit ci-dessus, l'entrée du LSP de récupération va propager l'erreur à l'entrée du LSP, qui peut alors signaler la suppression du LSP de récupération.

Il est aussi possible au nœud initiateur de la suppression de retirer un LSP de récupération d'une manière autre que "en douceur". Dans ce cas, l'initiateur envoie un message PathTear vers l'aval et un message PathErr avec une indication "Confirmation" (code et valeur d'erreur réglés à zéro) et le fanion Path_State_Removed établi à (1) vers le nœud d'entrée du LSP. Cette manière de supprimer un nœud non d'entrée N'EST PAS RECOMMANDÉE parce que dans certains cas, elle peut résulter en la suppression du LSP protégé.

4.3.1 Traitement modifié d'objet NOTIFY_REQUEST

Un ensemble de règles parallèle est appliqué aux nœuds de branche et de fusion pour permettre au nœud de branche ou de fusion d'ajouter un objet NOTIFY_REQUEST aux messages Path et Resv des LSP, protégé et de récupération. Les nœuds de branche ajoutent des objets NOTIFY_REQUEST aux messages Path, et les nœuds de fusion ajoutent des objets NOTIFY_REQUEST aux messages Resv.

Quand un nœud fait un embranchement ou une fusion sur un LSP de récupération, le nœud DEVRAIT inclure un seul objet NOTIFY_REQUEST dans le message Path du LSP de récupération, dans le cas d'un nœud de branche, ou le message Resv du LSP de récupération, dans le cas d'un nœud de fusion. L'adresse du nœud notifiant DOIT être réglée à l'adresse du routeur du nœud traitant.

Les nœuds de branche et de fusion DEVRAIENT aussi ajouter un objet NOTIFY_REQUEST au LSP protégé. Pour les nœuds de branche, l'adresse du nœud notifiant est réglée à l'adresse utilisée dans l'objet Gabarit d'expéditeur du LSP de récupération associé. Pour les nœuds de fusion, l'adresse du nœud notifiant est réglée à l'adresse utilisée dans l'objet Session du LSP de récupération associé. Un objet NOTIFY_REQUEST ajouté en local DOIT être mentionné en premier dans un message sortant ; tout objet NOTIFY_REQUEST reçu DOIT alors être mentionné dans le message dans l'ordre de réception. Noter qu'il peut en résulter une pile (ou liste ordonnée) d'objets.

Les procédures normales de notification sont alors suivies pour le LSP à protéger. C'est-à-dire, le nœud notificateur DOIT produire un message Notify au receveur indiqué par l'adresse de notification du premier objet NOTIFY_REQUEST mentionné. Sous le contrôle de la politique locale, un nœud qui produit un message Notify PEUT aussi envoyer un message Notify à l'adresse de nœud de notification indiquée dans le dernier, ou tout autre, objet NOTIFY_REQUEST porté dans le message Path ou Resv.

Les nœuds de fusion et de branche de LSP de récupération suppriment l'objet ajouté par le nœud de branche ou fusion de récupération provenant des messages Path et Resv sortants pour le LSP protégé. Ceci est fait en retirant l'objet NOTIFY_REQUEST qui, dans le cas d'un nœud de fusion, correspond à l'adresse de source du LSP de récupération et, dans le cas d'un nœud de branche, correspond à l'adresse de point d'extrémité de tunnel du LSP de récupération. L'objet NOTIFY_REQUEST correspondant va normalement être le premier de la liste des objets NOTIFY_REQUEST. Noter que pour couvrir certains scénarios de rétro compatibilité, l'objet NOTIFY_REQUEST NE DEVRAIT PAS être supprimé si il est le seul objet NOTIFY_REQUEST.

Noter que ceci exige le changement suivant au paragraphe 4.2.1 de la [RFC3473] :

ancien texte : Si un message contient plusieurs objets NOTIFY_REQUEST, seul le premier objet est significatif. Les objets NOTIFY_REQUEST suivants PEUVENT être ignorés et NE DEVRAIENT PAS être propagés.

nouveau texte : Si un message contient plusieurs objets NOTIFY_REQUEST, seul le premier objet est utilisé pour fournir les informations utilisées pour construire et envoyer une notification. Les objets NOTIFY_REQUEST suivants DOIVENT être propagés dans l'ordre reçu.

4.3.2 Traitement modifié de message Notify et Error

Les nœuds de branche et de fusion DOIVENT prendre en charge la modification suivante du traitement du message Notify.

Quand un nœud de branche ou de fusion reçoit la notification d'une défaillance de LSP et qu'il est incapable de récupérer de cette défaillance, il DOIT le notifier au nœud indiqué dans le premier objet NOTIFY_REQUEST reçu dans le message Path (pour les nœuds de branche) ou le message Resv (pour les nœuds de fusion) associés au LSP.

5. Objets Chemin d'enregistrement secondaire

Les objets d'enregistrement de chemin secondaire (SRRO, *Secondary Record Route Object*) sont utilisés pour enregistrer le chemin utilisé par les LSP de récupération.

5.1 Format

Le format d'un objet SECONDARY_RECORD_ROUTE est le même que celui d'un objet RECORD_ROUTE, numéro de classe 21. Cela inclut la définition des sous objets définis pour l'objet RECORD_ROUTE. La classe de l'objet SECONDARY_RECORD_ROUTE est 201 (de forme 11bbbbbb).

Le sous objet Protection défini ci-dessus peut aussi être utilisé dans les objets SECONDARY_RECORD_ROUTE.

5.2 Traitement de Path

Les SRRO peuvent être portés dans les messages Path et indiquer la présence de LSP de récupération en amont. Plus d'un SRRO PEUT être ajouté et être présent dans un message Path.

Tout SRRO reçu DOIT être transmis par les nœuds de transit, sans modification, dans le message Path sortant correspondant.

Les SRRO sont insérés dans les messages Path par les nœuds de fusion du LSP de récupération. Le SRRO est créé en copiant le contenu d'un RRO reçu par le LSP de récupération dans un nouvel objet SRRO. Ce SRRO est ajouté au message Path sortant du LSP récupéré. Noter que plusieurs SRRO peuvent être présents. La collection des SRRO est contrôlée via la présence d'un RRO dans les messages traités.

5.3 Traitement de Resv

Les SRRO peuvent être portés dans des messages Resv et indiquer la présence de LSP de récupération en aval. Plus d'un SRRO PEUT être ajouté et être présent dans un message Resv.

Tout SRRO reçu DOIT être transmis par les nœuds de transit, sans modification, dans le message Resv sortant correspondant. Quand des messages Resv sont fusionnés, le Resv fusionné devrait contenir tous les SRRO reçus dans les messages Resv en aval.

Les SRRO sont insérés dans les messages Resv par les nœuds de branche des LSP de récupération. Le SRRO DEVRAIT être créé avec les deux premiers objets qui sont l'adresse de nœud local, suivie par un sous objet Protection avec le contenu de l'objet PROTECTION du LSP de récupération. Le reste du SRRO DEVRAIT être créé en copiant le contenu du RRO reçu par le LSP de récupération. Ce SRRO DEVRAIT être ajouté au message Resv sortant du LSP récupéré. Là encore, plusieurs SRRO peuvent être présents.

Si le nouveau SRRO ajouté est cause que le message est trop gros pour tenir dans un message Resv, les sous objets SRRO DEVRAIENT être supprimés de tous les SRRO présents. Quand on supprime des sous objets, les deux premiers sous objets et le dernier sous objet dans un SRRO NE DOIVENT PAS être supprimés. Noter que le sous objet qui suivait un sous objet supprimé DOIT être mis à jour avec le bit L établi (1). Si, après la suppression de tous les sous objets sauf le premier et le dernier dans tous les SRRO, le message résultant est encore trop gros pour tenir, alors tous les SRRO DEVRAIENT être supprimés jusqu'à ce que le message tienne.

6. Contrôle dynamique de récupération de segment de LSP

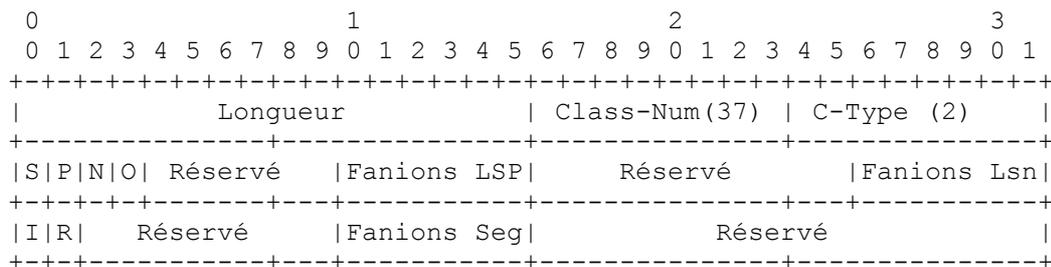
L'identification dynamique des nœuds de branche et de fusion est prise en charge via les fanions de récupération de

segment de LSP portés dans l'objet PROTECTION. Les fanions de récupération de segment de LSP sont portés dans un des champs Réserve définis dans l'objet PROTECTION défini dans la [RFC4872]. Les fanions de récupération de segment de LSP sont utilisés pour indiquer quand la récupération de segment de LSP est désirée. Quand ces bits sont établis, les nœuds de branche et de fusion sont identifiés de façon dynamique.

Noter que les procédures définies dans cette section sont parallèles aux procédures de contrôle explicite définies ci-dessus au paragraphe 4.2. La principale différence est dans la création d'un ERO de LSP de récupération.

6.1 Format d'objet PROTECTION modifié

Les fanions de récupération de segment de LSP sont portés dans l'objet PROTECTION de type C 2 défini dans la [RFC4872]. Le format de l'objet PROTECTION modifié portant ces fanions est :



Bit I (en place) : 1 bit. Quand il est établi (1) il indique que le type de récupération de segment désiré indiqué dans le fanion Récupération de segment de LSP est déjà en place pour le LSP associé.

Bit R (Requis) : 1 bit. Quand il est établi (1) il indique que la défaillance à établir la protection indiquée devrait résulter en une défaillance du LSP protégé.

Fanions Seg : fanions Récupération de segment, 6 bits. Ce champ est utilisé pour indiquer quand un nœud en amont désire que la récupération de segment de LSP soit initiée de façon dynamique lorsque possible. Les valeurs utilisées dans ce champ sont identiques aux valeurs définies pour les fanions de LSP ; voir la [RFC4872].

Voir la [RFC4872] pour la définition des autres champs.

6.2 Procédures de contrôle dynamique

Les fanions de récupération de segment de LSP sont réglés pour indiquer que la récupération de segment de LSP est désirée pour le LSP signalé. Le type de récupération désiré est indiqué par les fanions. La décision d'établir les fanions de récupération de segment de LSP est une affaire locale et sort du domaine d'application du présent document. Une valeur de zéro (0) signifie qu'aucune identification dynamique de nœuds de branche de récupération de segment n'est nécessaire pour le LSP associé. Quand le bit I (en-place) est établi, il signifie que le type désiré de récupération est déjà en place pour ce LSP particulier.

Un nœud de transit qui reçoit un message Path contenant un objet PROTECTION avec une valeur non zéro des fanions de récupération de segment de LSP et le bit en-place à zéro (0) DEVRAIT considérer si il peut prendre en charge le type de récupération indiqué et si il peut identifier un nœud de fusion approprié pour un LSP de récupération. L'identification dynamique NE DOIT PAS être faite quand le nœud traitant est identifié comme nœud de branche dans un SERO. Si un nœud est incapable de fournir le type indiqué de récupération ou d'identifier un nœud de fusion, le message Path DOIT être traité normalement, et les fanions de récupération de segment de LSP NE DOIVENT PAS être modifiés.

Quand un nœud s'identifie dynamiquement lui-même comme nœud de branche et identifie le nœud de fusion pour le type de récupération indiqué dans les fanions de récupération de segment de LSP, il tente d'établir un LSP de récupération. Les informations identifiées dynamiquement, avec le message Path du LSP à récupérer, sont utilisées pour créer le LSP de récupération.

Le message Path pour le LSP de récupération est créé au nœud de branche en clonant les objets portés dans le message Path entrant du LSP à protéger. Certains objets sont remplacés ou modifiés dans le message Path sortant du LSP de récupération. L'objet Sender_template DOIT être mis à jour pour utiliser une adresse (dans son champ Adresse d'expéditeur du tunnel) sur le nœud local, et l'identifiant de LSP DOIT être mis à jour pour assurer son unicité. L'objet Session DOIT

être mis à jour pour utiliser l'adresse du nœud de fusion identifié dynamiquement comme adresse de point d'extrémité de tunnel, l'identifiant de tunnel PEUT être mis à jour, et l'identifiant étendu de tunnel DOIT être réglé à l'adresse locale du nœud. L'objet PROTECTION est mis à jour avec le bit en-place établi (1). Aucun RRO et ERO présent dans le message Path entrant NE DOIT PAS être inclus dans le LSP de récupération. Un nouvel ERO PEUT être créé sur la base de toute information de chemin calculée dynamiquement par le nœud local.

Le message Path résultant est utilisé pour créer le LSP de récupération. Lorsque le LSP de récupération existe, l'objet PROTECTION dans le message Path original (sauf si il est outrepassé par la politique locale) DOIT aussi être mis à jour avec le bit en-place établi (1). À partir de ce point, le traitement standard de message Path est utilisé pour le traitement des messages Path, résultant et original.

Le nœud de fusion d'un LSP de récupération contrôlé de façon dynamique DEVRAIT remettre à zéro (0) le bit en-place dans l'objet PROTECTION du message Path sortant associé au LSP de récupération terminé.

À la différence du contrôle explicite, si la création d'un LSP de récupération identifié dynamiquement échoue pour une raison quelconque, le LSP de récupération est supprimé, et aucun message ou indication d'erreur n'est envoyé en amont. À cette exception près, toutes les autres procédures pour les LSP de récupération explicitement contrôlée s'appliquent aux LSP de récupération dynamiquement contrôlée. Ces autres procédures sont définies ci-dessus du paragraphe 4.2.1 au paragraphe 4.2.4.

7. Formats de message RSVP mis à jour

Cette Section présente les formats relatifs au message RSVP tel que modifié par le présent document. Lorsque ils diffèrent, les formats pour les LSP unidirectionnels sont présentés séparément des LSP bidirectionnels.

Le format d'un message Path est comme suit :

```
<Message Path> ::= <En-tête commun> [ <INTEGRITY> ]
    [ [ <MESSAGE_ID_ACK> | <MESSAGE_ID_NACK> ] ... ]
    [ <MESSAGE_ID> ]
    <SESSION> <RSVP_HOP>
    <TIME_VALUES>
    [ <EXPLICIT_ROUTE> ]
    <LABEL_REQUEST>
    [ <PROTECTION> ]
    [ <LABEL_SET> ... ]
    [ <SESSION_ATTRIBUTE> ]
    [ <NOTIFY_REQUEST> ... ]
    [ <ADMIN_STATUS> ]
    [ <ASSOCIATION> ... ]
    [ <SECONDARY_EXPLICIT_ROUTE> ... ]
    [ <POLICY_DATA> ... ]
    <descripteur d'envoyeur>
```

Le format de la description d'envoyeur pour les LSP unidirectionnels est :

```
<descripteur d'envoyeur> ::= <SENDER_TEMPLATE> <SENDER_TSPEC>
    [ <ADSPEC> ]
    [ <RECORD_ROUTE> ]
    [ <SUGGESTED_LABEL> ]
    [ <RECOVERY_LABEL> ]
    [ <SECONDARY_RECORD_ROUTE> ... ]
```

Le format de la description d'envoyeur pour les LSP bidirectionnels est :

```
<descripteur d'envoyeur> ::= <SENDER_TEMPLATE> <SENDER_TSPEC>
    [ <ADSPEC> ]
    [ <RECORD_ROUTE> ]
    [ <SUGGESTED_LABEL> ]
```

```
[ <RECOVERY_LABEL> ]
<UPSTREAM_LABEL>
[ <SECONDARY_RECORD_ROUTE> ... ]
```

Le format d'un message PathErr est comme suit :

```
<Message PathErr> ::= <En-tête commun> [ <INTEGRITY> ]
  [ [<MESSAGE_ID_ACK> | <MESSAGE_ID_NACK>] ... ]
  [ <MESSAGE_ID> ]
  <SESSION> <ERROR_SPEC>
  [ <ACCEPTABLE_LABEL_SET> ... ]
  [ <SECONDARY_EXPLICIT_ROUTE> ... ]
  [ <POLICY_DATA> ... ]
  <descripteur d'envoyeur>
```

Le format d'un message Resv est comme suit :

```
<En-tête commun> [ <INTEGRITY> ]
  [ [<MESSAGE_ID_ACK> | <MESSAGE_ID_NACK>] ... ]
  [ <MESSAGE_ID> ]
  <SESSION> <RSVP_HOP>
  <TIME_VALUES>
  [ <RESV_CONFIRM> ] [ <SCOPE> ]
  [ <NOTIFY_REQUEST> ... ]
  [ <ADMIN_STATUS> ]
  [ <POLICY_DATA> ... ]
  <STYLE> <liste des descripteurs de flux>
```

```
<liste des descripteurs de flux> ::= <liste des descripteurs de flux FF> | <descripteur de flux SE>
```

```
<liste des descripteurs de flux FF> ::= <FLOWSPEC> <FILTER_SPEC>
  <LABEL> [ <RECORD_ROUTE> ]
  [ <SECONDARY_RECORD_ROUTE> ... ]
  | <liste des descripteurs de flux FF>
  <FF descripteur de flux>
```

```
<descripteur de flux FF> ::= [ <FLOWSPEC> ] <FILTER_SPEC> <LABEL>
  [ <RECORD_ROUTE> ]
  [ <SECONDARY_RECORD_ROUTE> ... ]
```

```
<descripteur de flux SE> ::= <FLOWSPEC> <liste de spec de filtre SE>
```

```
<liste de spec de filtre SE> ::= <spec de filtre SE> | <liste de spec de filtre SE> <spec de filtre SE>
```

```
<spec de filtre SE> ::= <FILTER_SPEC> <LABEL> [ <RECORD_ROUTE> ]
  [ <SECONDARY_RECORD_ROUTE> ... ]
```

8. Considérations sur la sécurité

Le présent document introduit de nouveaux objets de message à utiliser dans la signalisation GMPLS [RFC3473]. Il n'introduit aucun nouveau message de signalisation, ni ne change les relations entre les LSR qui sont adjacents dans le plan de contrôle.

Les procédures définies dans le présent document résultent en un accroissement de la quantité d'informations topologiques portées dans les messages de signalisation car la présence des SERO et SRRO signifie nécessairement qu'il y a plus d'informations sur les chemins des LSP portés que dans les simples ERO et RRO. Donc, en cas d'interception d'un message de signalisation, il pourrait être déduit légèrement plus sur l'état du réseau que ce n'était précédemment le cas, mais ceci est jugé être un risque de sécurité très mineur car ces informations sont déjà disponibles via l'acheminement.

Autrement, le présent document n' introduit pas de considérations de sécurité supplémentaires. Voir dans la [RFC3473] les considérations de sécurité pertinentes.

9. Considérations relatives à l'IANA

L'IANA a alloué les valeurs suivantes pour les espaces de noms définis dans le présent document et repris dans cette section.

9.1 Allocation d'un nouveau type d'association

L'IANA a fait l'allocation suivante au registre "Paramètres de signalisation GMPLS" (voir la [RFC4872]) situé à <http://www.iana.org/assignments/gmpls-sig-parameters> .

Valeur	Type
2	Partage de ressource (R) [RFC4873]

9.2 Définition des bits réservés d'objet PROTECTION

Le présent document définit les bits portés dans le champ Réserve de l'objet PROTECTION défini dans la [RFC4872]. Comme aucun registre IANA n'est demandé pour ces bits dans la [RFC4872], aucune action de l'IANA n'est exigée à l'égard de cette définition.

9.3 Objet Chemin secondaire explicite

L'IANA a fait les allocations suivantes dans la section "Noms, numéros, et types de classes" du registre "Paramètres RSVP" situé à <http://www.iana.org/assignments/rsvp-parameters> .

Une nouvelle classe appelée SECONDARY_EXPLICIT_ROUTE a été créée dans la gamme 11bbbbbb (200) avec la définition suivante :

Types de classe ou C-types : mêmes valeurs que l'objet EXPLICIT_ROUTE (C-Num 20)

Pour Classe 1, C-Type 1, le type suivant de sous objet supplémentaire est défini :

37	PROTECTION	[RFC4873]
----	------------	-----------

9.4 Objet Chemin d'enregistrement secondaire

L'IANA a fait les allocations suivantes dans la section "Noms, numéros, et types de classes" du registre "Paramètres RSVP" situé à <http://www.iana.org/assignments/rsvp-parameters> .

Une nouvelle classe appelée SECONDARY_RECORD_ROUTE a été créée dans la gamme 11bbbbbb (201) avec la définition suivante :

Types de classe ou C-types : mêmes valeurs que l'objet RECORD_ROUTE (C-Num 21)

Pour la classe 1, C-Type 1, le type suivant de sous objet supplémentaire est défini :

37	PROTECTION	[RFC4873]
----	------------	-----------

9.5 Nouveau code d'erreur

L'IANA a fait les allocations suivantes dans la sous section "Problème d'acheminement" de la section "Codes et valeurs d'erreur" du registre "Paramètres RSVP" situé à <http://www.iana.org/assignments/rsvp-parameters> .

21 = Échec de protection de segment de LSP [RFC4873]

9.6 Utilisation du C-type d'objet PROTECTION

Le présent document modifie l'objet PROTECTION, numéro de classe 37, C-Type 2 (défini au paragraphe 14.1. de la [RFC4872]).

10. Références

10.1 Références normatives

- [RFC2119] S. Bradner, "[Mots clés à utiliser](#) dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997. (MàJ par [RFC8174](#))
- [RFC2205] R. Braden, éd., L. Zhang, S. Berson, S. Herzog, S. Jamin, "[Protocole de réservation de ressource](#) (RSVP) -- version 1, spécification fonctionnelle", septembre 1997. (MàJ par [RFC2750](#), [RFC3936](#), [RFC4495](#), [RFC6780](#)) (P.S.)
- [RFC3209] D. Awduche, et autres, "[RSVP-TE : Extensions à RSVP pour les tunnels LSP](#)", décembre 2001. (Mise à jour par [RFC3936](#), [RFC4420](#), [RFC4874](#), [RFC5151](#), [RFC5420](#), [RFC6790](#))
- [RFC3471] L. Berger, éd., "[Commutation d'étiquettes multi-protocoles généralisée](#) (GMPLS) : description fonctionnelle de la signalisation", janvier 2003. (MàJ par [RFC4201](#), [RFC4328](#), [RFC4872](#), [RFC8359](#)) (P.S.)
- [RFC3473] L. Berger, "[Extensions d'ingénierie de protocole](#) - trafic de signalisation de réservation de ressource (RSVP-TE) de commutation d'étiquettes multi-protocoles généralisée (GMPLS)", janvier 2003. (P.S., MàJ par 4003, 4201, 4420, 4783, 4784, 4873, 4974, 5063, 5151, [8359](#))
- [RFC4872] P. Lang et autres, "[Extensions RSVP-TE](#) pour la prise en charge de la récupération de commutation d'étiquettes multi protocoles généralisée (GMPLS) de bout en bout", mai 2007. (MàJ [RFC3471](#)) (MàJ par [RFC4873](#), [RFC6780](#)) (P.S.)

10.2 Références pour information

- [RFC4090] P. Pan et autres, "[Extensions de réacheminement rapide à RSVP-TE](#) pour les tunnels de LSP", mai 2005. (P.S. ; MàJ par [RFC8271](#), [RFC8537](#), [RFC8796](#))
- [RFC4426] J. Lang et autres, "[Spécification fonctionnelle de récupération](#) du protocole généralisé de commutation d'étiquettes multiprotocoles (GMPLS)", mars 2006. (P.S.)
- [RFC4427] E. Mannie et autres, "[Terminologie de récupération \(protection et restauration\)](#) pour le protocole généralisé de commutation d'étiquettes multiprotocoles (GMPLS)", mars 2006. (Information)

Adresse des auteurs

Lou Berger
LabN Consulting, L.L.C.
téléphone : +1 301-468-9228
mél : lberger@labn.net

Igor Bryskin
ADVA Optical
7926 Jones Branch Drive
McLean VA, 22102
mél : IBryskin@advaoptical.com

Dimitri Papadimitriou
Alcatel
Francis Wellesplein 1
B-2018 Antwerpen, Belgium
téléphone : +32 3 240-8491

Adrian Farrel
Old Dog Consulting
téléphone : +44 (0) 1978 860944
mél : adrian@olddog.co.uk

mél : dimitri.papadimitriou@alcatel-lucent.be

Déclaration complète de droits de reproduction

Copyright (C) The IETF Trust (2007)

Le présent document est soumis aux droits, licences et restrictions contenus dans le BCP 78, et sauf pour ce qui est mentionné ci-après, les auteurs conservent tous leurs droits.

Le présent document et les informations 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, le IETF TRUST 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 encloses ne viole 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 faites 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 le 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 la Internet Society.