

Groupe de travail Réseau
Request for Comments : 2745
 Catégorie : En cours de normalisation

Traduction Claude Brière de L'Isle

A. Terzis, UCLA
 B. Braden, ISI
 S. Vincent, Cisco Systems
 L. Zhang, UCLA
 janvier 2000

Messages de diagnostic RSVP

Statut de ce mémoire

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

Notice de copyright

Copyright (C) The Internet Society (2000).

Résumé

Le présent document spécifie les facilités de diagnostic de RSVP, qui permettent à un utilisateur de collecter des informations sur l'état de RSVP le long d'un chemin. La présente spécification décrit la fonctionnalité, les formats de message de diagnostic, et les règles de traitement.

Table des Matières

1. Introduction.....	1
2. Vue d'ensemble.....	2
3. Format de paquet de diagnostic.....	3
3.1 En-tête commun de message RSVP.....	3
3.2 Objet RSVP_HOP de prochain bond.....	4
3.3 Objet DIAGNOSTIC.....	4
3.4 Objet DIAG_SELECT.....	5
3.5 Objet ROUTE.....	6
3.6 Objet DIAG_RESPONSE.....	6
3.7 Objet TUNNEL.....	7
4. Règles de transmission de paquet de diagnostic.....	7
4.1 Transmission de paquet DREQ.....	7
4.2. Transmission de paquet DREP.....	9
4.3 Choix et ajustement de la MTU.....	9
4.4 Erreurs.....	9
5. Diagnostic de problèmes en utilisant les facilités de diagnostic de RSVP.....	10
5.1 À travers des pare-feu.....	10
5.2 Examen des temporisateurs RSVP.....	10
5.3 Découverte de nuages non RSVP.....	10
5.4 Découverte de fusions de réservation.....	10
5.5 Diagnostics d'erreur.....	10
5.6 Traversée de routeurs RSVP "traditionnels".....	10
6. Commentaires sur la mise en œuvre de client de diagnostic.....	11
7. Considérations sur la sécurité.....	12
8. Remerciements.....	12
9. Références.....	12
10. Adresse des auteurs.....	12
10. Déclaration complète de droits de reproduction.....	13

1. Introduction

Dans le protocole RSVP de base [RFC2205], les messages d'erreur sont le seul moyen qu'a un hôte d'extrémité de recevoir des retours sur une défaillance de l'établissement de l'état du chemin ou de l'état de réservation. Un message d'erreur ne rapporte que les informations provenant du point défaillant, sans aucune information sur l'état des autres bonds avant ou

après la défaillance. En l'absence de défaillance, un hôte ne reçoit pas de retour concernant les détails d'une réservation qui a été mise en place, ni si, ou comment, sa propre demande de réservation a été fusionnée avec celle d'autres. De telles informations peuvent être très désirables pour des besoins de débogage, ou pour la gestion des ressources réseau en général.

Le présent document spécifie la facilité de diagnostic de RSVP, qui est conçue pour combler ce trou dans l'information. La facilité de diagnostic peut être utilisée pour collecter et rapporter les informations d'état de RSVP le long du chemin d'un receveur à un envoyeur spécifique. Il utilise les messages Diagnostic qui sont indépendants des autres messages de contrôle RSVP et ne produisent pas d'effets collatéraux ; c'est-à-dire, ils ne changent pas l'état RSVP chez les nœuds ou hôtes. De même, ils ne fournissent pas de rapport d'erreur mais plutôt une collection d'informations demandées d'état de RSVP.

La facilité de diagnostic de RSVP a été conçue avec les objectifs suivants :

- Collecter les informations d'état de RSVP sur chaque bond à capacité RSVP le long d'un chemin défini par l'état du chemin, soit d'une réservation existante, soit avant que soit faite une demande de réservation. Plus précisément, on veut être capable de collecter des informations sur les spécifications de flux (*flowspec*), les valeurs de temporisateur de rafraîchissement, et les fusions de réservation à chaque bond le long du chemin.
- Collecter le compte de bonds IP à travers chaque nuage non RSVP.
- Éviter l'implosion ou l'explosion de paquets de diagnostic.

Est spécifiquement identifié comme un non objectif :

- Vérifier la disponibilité de ressources le long d'un chemin. Une telle fonctionnalité pourrait être utile pour de futures demandes de réservation, mais cela exigerait des modifications des modules existants de contrôle d'admission qui sortent du domaine d'application de RSVP.

2. Vue d'ensemble

La facilité de diagnostic introduit deux nouveaux types de message RSVP : Demande de diagnostic (DREQ, *Diagnostic Request*) et Réponse de diagnostic (DREP, *Diagnostic Reply*). Un message DREQ peut être généré par un client dans un hôte "demandeur", qui peut être ou non un participant à la session RSVP sur laquelle faire le diagnostic. Un client dans l'hôte demandeur invoque la facilité de diagnostic de RSVP en générant un paquet DREQ et en l'envoyant vers le nœud LAST-HOP (*dernier bond*), qui devrait être sur le chemin RSVP à diagnostiquer. Ce paquet DREQ spécifie la session RSVP et un hôte envoyeur pour cette session. En commençant par le LAST-HOP, le paquet DREQ collecte les informations bond par bond lorsque il est transmis vers l'envoyeur (voir la Figure 1) jusqu'à ce qu'il atteigne le nœud terminal. Spécifiquement, chaque bond à capacité RSVP ajoute au message DREQ un objet de réponse (DIAG_RESPONSE) contenant l'état RSVP local pour la session RSVP spécifiée.

Lorsque le paquet DREQ atteint le nœud d'extrémité, le type de message est changé en réponse de diagnostic (DREP, *Diagnostic Reply*) et la réponse complétée est envoyée au nœud demandeur d'origine. Des réponses partielles peuvent aussi être retournées avant que le paquet DREQ atteigne le nœud d'extrémité si une condition d'erreur sur le long du chemin, comme un "pas d'état du chemin", empêche la suite de la transmission du paquet DREQ. Pour éviter l'implosion ou l'explosion de paquets, tous les paquets de diagnostic sont uniquement en envoi individuel.

Donc, il y a généralement trois nœuds (hôtes et/ou routeurs) impliqués dans l'exécution de la fonction de diagnostic : le nœud demandeur, le nœud de début, et le nœud d'extrémité, comme le montre la Figure 1. Il est possible que le client qui invoque la fonction de diagnostics puisse résider directement sur le nœud de début, et dans ce cas les deux premiers nœuds sont le même. Le nœud de début est appelé "LAST-HOP", signifiant le dernier bond du segment de chemin à diagnostiquer. Le nœud LAST-HOP peut être un nœud receveur ou un nœud intermédiaire le long du chemin. Le nœud d'extrémité est généralement l'hôte envoyeur spécifié. Cependant, le client peut limiter la longueur du segment de chemin à diagnostiquer en spécifiant une limite du compte de bonds dans le message DREQ.

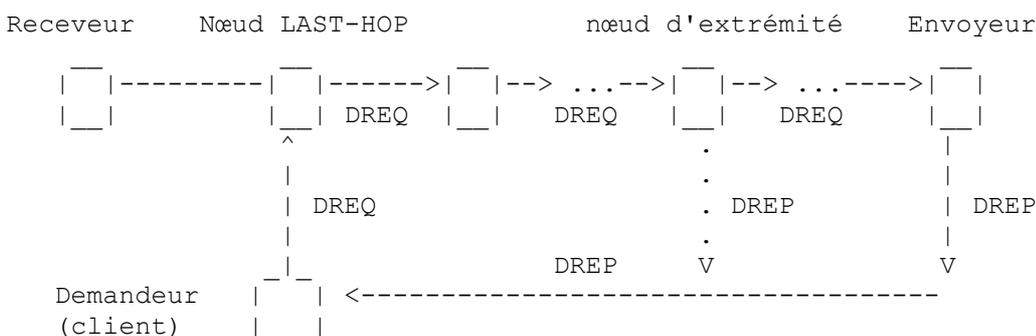


Figure 1

Les paquets DREP peuvent être en envoi individuel à partir du nœud d'extrémité et retour vers le demandeur directement ou bond par bond le long du chemin inverse de celui pris par le message DREQ jusqu'au LAST-HOP, et de là au demandeur. Le retour direct est plus rapide et plus efficace, mais le chemin inverse bond par bond peut être le seul choix si les paquets doivent traverser des pare-feu. Le retour bond par bond est réalisé en utilisant un objet ROUTE facultatif, qui est construit de façon incrémentaire pour contenir une liste des adresses de nœuds par lesquels le paquet DREQ est passé. L'objet ROUTE est alors utilisé en sens inverse comme chemin de source pour transmettre le DREP bond par bond en retour au nœud LAST-HOP.

Un message DREQ consiste toujours en un seul datagramme IP non fragmenté. Par ailleurs, un message DREQ peut générer plusieurs paquets DREP, contenant chacun un fragment du message DREQ total. Lorsque le chemin consiste en de nombreux bonds, la longueur totale d'un message DREP va excéder la taille de MTU avant d'atteindre le nœud d'extrémité ; donc, le message doit être fragmenté. S'appuyer sur la fragmentation et le réassemblage IP peut, cependant, être problématique, en particulier lorsque les messages DREP sont retournés au demandeur bond par bond, cas dans lequel la fragmentation/réassemblage devrait être effectuée à chaque bond. Pour éviter des frais généraux excessifs, on laisse le demandeur définir une taille de MTU de chemin par défaut qui est portée dans chaque paquet DREQ. Si un nœud intermédiaire trouve que la taille de la MTU par défaut est plus grosse que la MTU de l'interface entrante, il réduit la taille de MTU par défaut à la taille de MTU de l'interface entrante. Si un nœud intermédiaire détecte qu'un paquet DREQ est d'une taille qui dépasse celle de la MTU par défaut, il retourne au demandeur (d'une des manières décrites ci-dessus) un fragment DREP contenant les réponses accumulées. Il retire ensuite ces réponses de la DREQ et continue de la transmettre. Le nœud demandeur peut réassembler les fragments de DREP résultants en un message DREP complet.

Lorsque il discute du traitement d'un paquet de diagnostic, le présent document utilise une terminologie de direction qui est cohérente avec la spécification fonctionnelle de RSVP [RFC2205], relativement à la direction des flux de paquets de données. Donc, un paquet DREQ entre dans un nœud par une "interface sortante" et est transmis vers l'expéditeur par une "interface sortante", parce que les paquets DREQ voyagent en sens inverse du flux de données.

Remarquer que les paquets DREQ ne peuvent être transmis qu'après que l'état du chemin RSVP a été établi. Si aucun état du chemin n'existe, on peut revenir à la facilité traceroute ou mtrace pour examiner si l'acheminement individuel/diffusion groupée fonctionne correctement.

3. Format de paquet de diagnostic

Comme les autres messages RSVP, les messages DREQ et DREP consistent en un en-tête RSVP commun suivi par un ensemble variable d'objets de données RSVP typés. On doit utiliser la séquence suivante :

```
+-----+
| En-tête commun de message RSVP |
+-----+
|      Objet de session            |
+-----+
|  Objet RSVP_HOP de prochain bond |
+-----+
|      Objet DIAGNOSTIC           |
+-----+
|  Objet (facultatif) DIAG_SELECT |
+-----+
|  Objet (facultatif) ROUTE       |
+-----+
|zéro ou plusieurs objets DIAG_RESPONSE|
+-----+
```

L'objet de session identifie la session RSVP pour laquelle les informations d'état sont collectées. On décrit chacune des autres parties.

3.1 En-tête commun de message RSVP

L'en-tête commun de message RSVP est défini dans la [RFC2205]. Les exceptions et extensions spécifiques suivantes sont nécessaires pour DREP et DREQ.

Champ Type : on définit :

Type = 8 : DREQ (*Diagnostic Request*) demande de diagnostic

Type = 9 : DREP (*Diagnostic Reply*) réponse de diagnostic

Longueur RSVP : Si c'est un message DREP et si le fanion MF dans l'objet DIAGNOSTIC (voir ci-dessous) est établi, ce champ indique la longueur de ce seul fragment DREP plutôt que la longueur totale du message DREP de réponse complet (qui ne peut généralement pas être connue à l'avance).

3.2 Objet RSVP_HOP de prochain bond

Cet objet RSVP_HOP porte le "traitement d'interface logique" (LIH, *Logical Interface Handle*) de l'interface à travers laquelle la DREQ devrait être reçue au nœud amont. Cet objet est mis à jour bond par bond. Il est utilisé pour les mêmes raisons qu'un message RESV contient un objet RSVP_HOP : pour distinguer les interfaces logiques et éviter les problèmes causés par les asymétries d'acheminement et les nuages non RSVP.

Bien que l'adresse IP ne soit pas réellement utilisée durant le traitement de DREQ, par cohérence avec l'utilisation de l'objet RSVP_HOP dans d'autres messages RSVP, l'adresse IP dans l'objet RSVP_HOP va contenir l'adresse de l'interface à travers laquelle la DREQ a été envoyée.

3.3 Objet DIAGNOSTIC

Un objet DIAGNOSTIC contient les informations communes de contrôle de diagnostic dans les deux messages DREQ et DREP.

o objet DIAGNOSTIC IPv4 : Classe = 30, C-Type = 1

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|Max bonds RSVP |Comp.  bond RSVP|          Réserve          |MF|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     Identifiant de demande                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|          MTU du chemin          |          Décalage de fragment          |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     Adresse du dernier bond                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     Objet SENDER_TEMPLATE                                     |
|                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     Objet FILTER_SPEC du demandeur                                     |
|                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

Ici, toutes les adresses IP utilisent le format IPv4 à quatre octets, explicitement dans l'adresse de dernier bond et en utilisant les formes IPv4 des objets incorporés FILTER_SPEC et RSVP_HOP.

o objet DIAGNOSTIC IPv6 : Classe = 30, C-Type = 2

Le format est le même, excepté que toutes les adresses IP explicites et incorporées sont des adresses IPv6 de 16 octets. Les champs sont les suivants :

Max bonds RSVP : Un octet qui spécifie le nombre maximum de bonds RSVP sur lesquels l'information sera collectée. Si une condition d'erreur au milieu du chemin empêche le paquet DREQ d'atteindre le nœud d'extrémité spécifié, le champ Max bonds RSVP peut être utilisé pour effectuer une recherche de longueur d'expansion pour atteindre le point juste avant le problème. Si cette valeur est 1, le nœud de début et le nœud d'extrémité de l'interrogation sont le même. Si c'est zéro, il n'y a pas de limite au nombre de bonds.

Compte de bonds RSVP : Enregistre le nombre de bonds RSVP qui ont été traversés jusqu'à présent. Si les nœuds de début et de fin sont le même, cette valeur sera 1 dans le message DREP résultant.

Décalage de fragment : Indique si ce fragment DREP appartient au message DREP complet, mesuré en octets. Le premier fragment a le décalage zéro. Décalage de fragment est utilisé aussi pour déterminer si un message DREQ contenant zéro objet DIAG_RESPONSE devrait être traité à un nœud à capacité RSVP.

Fanion MF : Ce fanion signifie "plus de fragments". Il doit être réglé à zéro (0) dans tous les messages DREQ. Il doit être réglé à un (1) dans tous les paquets DREP qui portent des résultats partiels et sont retournés par des nœuds intermédiaires à cause de la limite de MTU. Lorsque le message DREQ est converti en un message DREP dans le nœud d'extrémité, la fanion MF doit rester à zéro.

Identifiant de demande : Identifie un message DREQ individuel dans le message DREP correspondant (ou tous les fragments du message de réponse). Une façon possible de définir l'identifiant de demande serait d'utiliser 16 bits pour spécifier l'identifiant du processus constituant l'interrogation et 16 bits pour distinguer les différentes interrogations de ce processus.

MTU du chemin : Spécifie une taille de MTU par défaut en octets pour les messages DREP et DREQ. Cette valeur ne devrait pas être inférieure à la taille du paquet DREQ de "base". Un paquet DREQ de "base" est celui qui contient un en-tête commun, un objet Session, un objet RSVP_HOP de dernier bond, un objet DIAGNOSTIC, un objet ROUTE vide et une seule DIAG_RESPONSE par défaut (voir ci-dessous). L'hypothèse faite ici est qu'un paquet de diagnostic de cette taille peut toujours être transmis sans fragmentation IP.

Adresse de dernier bond : Adresse IP du nœud LAST-HOP. Le message DREQ commence par collecter l'information à ce nœud et continue en remontant vers l'expéditeur.

Objet SENDER_TEMPLATE : Cet objet IPv4/IPv6 SENDER_TEMPLATE contient l'adresse IP et l'accès d'un expéditeur pour la session objet du diagnostic. Le paquet DREQ est transmis bond par bond jusqu'à cette adresse.

Objet FILTER_SPEC du demandeur : Cet objet IPv4/IPv6 FILTER_SPEC contient l'adresse IP et l'accès à partir desquels la demande a été générée et auxquels le ou les messages DREP devraient être envoyés.

3.4 Objet DIAG_SELECT

o DIAG_SELECT : Classe = 33, C-Type = 1.

Un message Diagnostic peut facultativement contenir un objet DIAG_SELECT pour spécifier quels objets RSVP spécifiques devraient être rapportés dans un objet DIAG_RESPONSE. En l'absence d'un objet DIAG_SELECT, l'objet DIAG_RESPONSE ajouté par le nœud contiendra un ensemble par défaut de types d'objets (voir l'objet DIAG_RESPONSE ci-dessous).

L'objet DIAG_SELECT contient une liste de paires [Classe, C-type], au format suivant :

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| classe | C-Type | classe | C-Type |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
//                                                                 //
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| classe | C-Type | classe | C-Type |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

Lorsque un objet DIAG_SELECT est inclus dans un message DREQ, chaque nœud RSVP le long du chemin va ajouter un objet DIAG_RESPONSE contenant les objets de réponse (voir ci-dessous) dont la classe et le C-Type correspondent aux entrées dans la liste DIAG_SELECT (et sont du chemin et de l'état de réservation correspondants). Un octet C-type de zéro est un "caractère générique" (*wildcard*) correspondant à tout C-Type associé à la classe associée.

Selon le type des objets demandés, un nœud peut trouver les informations associées dans l'état de chemin ou de réservation mémorisé pour la session décrite dans l'objet SESSION. Précisément, les informations pour les objets RSVP_HOP, SENDER_TEMPLATE, SENDER_TSPEC, ADSPEC peuvent être extraites de l'état du chemin du nœud, tandis que les informations pour les objets FLOWSPEC, FILTER_SPEC, CONFIRM, STYLE et SCOPE peuvent être trouvées dans l'état de réservation du nœud (si ils existent).

Si le nombre de paires [Classe, C-Type] est impair, les deux derniers octets de l'objet DIAG_SELECT doivent être zéro. Un objet DIAG_SELECT maximum est celui qui contient les paires [Classe, C-type] pour tous les objets RSVP qui peuvent être demandés dans une interrogation Diagnostic.

3.5 Objet ROUTE

Un message diagnostic peut contenir un objet ROUTE, qui est utilisé pour enregistrer le chemin du message DREQ et comme route de source pour retourner le ou les messages DREP bond par bond.

- o Objet IPv4 ROUTE : Classe = 31, C-Type = 1.

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|          réservé          | Pointeur R |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               |           |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               |           |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Cet objet signifie comment la réponse devrait être retournée. Si il n'existe pas dans le paquet DREQ, les paquets DREP devraient être envoyés directement au demandeur. Si il existe bien, les paquets DREP doivent être retournés bond par bond le long du chemin inverse au nœud LAST-HOP et de là au nœud demandeur.

Un objet ROUTE vide est celui qui a une liste vide de nœuds RSVP et dont le pointeur R est égal à zéro.

Liste de nœuds RSVP : Liste d'adresses IPv4 de nœuds RSVP. Le nombre d'adresses dans cette liste peut être calculé à partir de la taille de l'objet.

Pointeur R : Utilisé seulement dans les messages DREP (voir les détails au paragraphe 4.2) mais il est incrémenté lorsque chaque bond ajoute son adresse d'interface entrante dans l'objet ROUTE.

- o Objet IPv6 ROUTE : Classe = 31, C-Type = 2

Même chose, sauf que la liste de nœuds RSVP contient des adresses IPv6.

Dans un message DREQ, la liste des nœuds RSVP spécifie tous les bonds RSVP entre l'adresse de dernier bond spécifiée dans l'objet DIAGNOSTIC, et le dernier nœud RSVP que le message DREQ a visité. Dans un message DREP, la liste des nœuds RSVP spécifie tous les bonds RSVP entre le dernier bond et le nœud qui retourne ce message DREP.

3.6 Objet DIAG_RESPONSE

Chaque nœud RSVP attache un objet DIAG_RESPONSE à chaque message DREQ qu'il reçoit, avant de transmettre le message. L'objet DIAG_RESPONSE contient l'état à rapporter pour ce nœud. Il a un en-tête de format fixé et ensuite une liste variable d'objets d'état RSVP, ou "objets de réponse".

- o Objet IPv4 DIAG_RESPONSE : Classe = 32, C-Type = 1.

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|          Heure d'arrivée de DREQ          |           |
+-----+-----+-----+-----+-----+-----+-----+-----+
|          Adresse d'interface entrante          |           |
+-----+-----+-----+-----+-----+-----+-----+-----+
|          Adresse d'interface sortante          |           |
+-----+-----+-----+-----+-----+-----+-----+-----+
|          Adresse du routeur du bond RSVP précédent          |           |
+-----+-----+-----+-----+-----+-----+-----+-----+
|  D-TTL      |M|R-err|  K      |  Valeur de temporisateur      |           |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               |           |
|          Objet TUNNEL (facultatif)          |           |
|                               |           |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                               |           |
//                               Objets Réponse                               //
|                               |           |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

- o Objet IPv6 DIAG_RESPONSE : Classe = 32, C-Type = 2.

Cet objet a le même format, excepté que toutes les adresses IP explicites et incorporées sont des adresses IPv6.

Les champs sont les suivants :

Heure d'arrivée de DREQ : horodatage NTP de 32 bits qui spécifie l'heure d'arrivée du message DREQ à ce nœud. La forme de 32 bits d'un horodatage NTP consiste en les 32 bits du milieu de la forme complète à 64 bits, c'est-à-dire, les 16 bits de moindre poids de la partie entière et les 16 bits de poids fort de la partie fractionnaire.

Adresse d'interface entrante : spécifie l'adresse IP de l'interface sur laquelle les messages provenant de l'expéditeur sont supposés arriver, ou 0 si ce n'est pas connu.

Adresse d'interface sortante : spécifie l'adresse IP de l'interface par laquelle le message DREQ est arrivé et à laquelle les messages provenant d'un certain expéditeur et pour l'adresse de session spécifiée s'écoulent, ou 0 si ce n'est pas connu.

Adresse du routeur du bond RSVP précédent : spécifie l'adresse IP de laquelle ce nœud reçoit les messages RSVP PATH pour cette source, ou 0 si c'est inconnu. C'est aussi l'interface à laquelle la DREQ sera transmise.

D-TTL : nombre de bonds IP traversés par ce message DREQ depuis le nœud RSVP aval jusqu'au nœud actuel.

Fanion M : fanion d'un seul bit qui indique si la réservation décrite par les objets de réponse est fusionnée avec des réservations provenant d'autres interfaces en aval lorsque elle est transmise vers l'amont.

R-erreur : champ de trois bits qui indique une condition d'erreur à un nœud. Les valeurs actuellement définies sont :

0x00 : pas d'erreur

0x01 : pas d'état du chemin

0x02 : paquet trop gros

0x04 : objet ROUTE trop gros

K : valeur de multiple de temporisateur de rafraîchissement (défini au paragraphe 3.7 de la [RFC2205]).

Valeur de temporisateur : valeur du temporisateur de rafraîchissement local en secondes.

L'ensemble d'objets de réponse à inclure à la fin de l'objet DIAG_RESPONSE est déterminé par un objet DIAG_SELECT, si il en est un présent. Si aucun objet DIAG_SELECT n'est présent, les objets de réponse appartiennent à la liste de classes par défaut : objet SENDER_SPEC, objet FILTER_SPEC, objet FLOWSPEC, objet STYLE.

Tout C-Type présent dans l'état RSVP local sera utilisé. Ces objets de réponse peuvent être dans tout ordre mais ils doivent tous être à la fin de l'objet DIAG_RESPONSE.

Un objet DIAG_RESPONSE par défaut est celui qui contient la liste de classes par défaut décrite ci-dessus.

3.7 Objet TUNNEL

L'objet facultatif TUNNEL devrait être inséré lorsque un message DREQ arrive à un nœud RSVP qui agit comme point de sortie de tunnel.

L'objet TUNNEL fournit la transposition entre la session RSVP de bout en bout qui est diagnostiquée et la session RSVP de l'autre côté du tunnel. Ces informations de transposition permettent au client de diagnostic de conduire le diagnostic sur la session tunnel impliquée, en invoquant une interrogation Diagnostic séparée pour la session tunnel et l'expéditeur tunnel correspondants. On se souviendra cependant que plusieurs sessions de bout en bout peuvent toutes se transposer en une session tunnel préconfigurée qui peut avoir des réglages de paramètres totalement différents.

L'objet tunnel est défini dans la spécification de tunnel RSVP [RFC2746].

4. Règles de transmission de paquet de diagnostic

4.1 Transmission de paquet DREQ

Les messages DREQ sont transmis bond par bond via envoi individuel de l'adresse LAST-HOP à l'adresse de l'expéditeur, comme spécifié dans l'objet DIAGNOSTIC. Si un nœud à capacité RSVP, autre que le nœud LAST-HOP, reçoit un message DREQ qui ne contient pas d'objet DIAG_RESPONSE et a un décalage de fragment de zéro, le nœud devrait transmettre le

paquet DREQ au LAST-HOP sans faire aucun des traitements mentionnés ci-dessous. La raison en est que de telles conditions ne s'appliquent que pour les nœuds en aval du LAST-HOP où aucune information ne devrait être collectée.

Le traitement commence lorsque un message DREQ, DREQ_in, arrive à un nœud.

1. Créer un nouvel objet DIAG_RESPONSE. Calculer le compte de bonds IP à partir du bond RSVP précédent. On fait cela en soustrayant la valeur du TTL dans l'en-tête IP de celle de Send_TTL dans l'en-tête RSVP commun. On sauvegarde le résultat dans le champ D-TTL de l'objet DIAG_RESPONSE.
2. Régler l'heure d'arrivée DREQ et l'adresse d'interface sortante dans l'objet DIAG_RESPONSE. Si ce nœud est le LAST-HOP, alors le champ Adresse d'interface entrante dans l'objet DIAG_RESPONSE contient la valeur suivante selon la session diagnostiquée :
 - * si la session en question est une session en envoi individuel, le champ Adresse d'interface sortante contient l'adresse de l'interface qu'utilise LAST-HOP pour envoyer les messages et données PATH au receveur spécifié par l'adresse de session ;
 - * autrement, si c'est une session en diffusion groupée et si il y a au moins un receveur pour cette session, LAST_HOP devrait utiliser l'adresse d'une des interfaces locales utilisées pour joindre un des receveurs ;
 - * autrement, l'adresse d'interface sortante devrait être zéro.
3. Incrémenter le champ Compte de bonds RSVP dans l'objet message DIAGNOSTIC de un.
4. Si aucun état du chemin n'existe pour la session spécifiée, régler R-erreur = 0x01 (pas d'état du chemin) et passer à l'étape 7.
5. Régler le reste des champs dans l'objet DIAG_RESPONSE. Si DREQ_in contient un objet DIAG_SELECT, les classes d'objet de réponse sont celles spécifiées dans le DIAG_SELECT ; autrement, elles sont les objets SENDER_TSPEC, STYLE, et FLOWSPEC. Si aucun état de réservation n'existe pour la session RSVP spécifiée, l'objet DIAG_RESPONSE ne va contenir aucun objet FLOWSPEC, FILTER_SPEC ou STYLE. Si il n'existe ni PATH ni état de réservation pour la session RSVP spécifiée, aucun objet de réponse ne sera ajouté à l'objet DIAG_RESPONSE.
6. Si le compte de bonds RSVP fait moins que Max-bonds-RSVP et si ce nœud n'est pas l'expéditeur, alors la DREQ est éligible à la transmission ; régler la MTU du chemin au minimum de la MTU de chemin et de la taille de MTU de l'interface entrante pour l'expéditeur objet du diagnostic.
7. Si la taille de DREQ_in plus la taille du nouvel objet DIAG_RESPONSE plus la taille d'une adresse IP (si il existe un objet ROUTE et si R-erreur= 0) est supérieure à la MTU du chemin, alors le nouveau message de diagnostic va être trop gros pour la transmission ou être retourné sans fragmentation ; régler le bit d'erreur "paquet trop gros" (0x02) dans la DIAG_RESPONSE et passer à l'étape SD1 dans Send_DREP (ci-dessous).
8. Si le bit d'erreur "Pas d'état du chemin" (0x01) est établi ou si le compte de bonds RSVP est égal à Max-bonds-RSVP ou si ce nœud est l'expéditeur, alors la DREQ ne peut pas être transmise plus loin ; passer à l'étape 10.
9. Transmettre la DREQ vers l'expéditeur, comme suit. Si un objet ROUTE existe, ajouter "Adresse d'interface entrante" à la fin de l'objet ROUTE et incrémenter le pointeur R de un. Mettre à jour l'objet RSVP_HOP de prochain bond, ajouter le nouvel objet DIAG_RESPONSE à la liste des objets DIAG_RESPONSE, et mettre à jour le champ Longueur du message dans l'en-tête RSVP commun en conséquence. Finalement, recalculer la somme de contrôle, transmettre DREQ_in au prochain bond vers l'expéditeur, et revenir.
10. Transformer la DREQ en DREP et retourner au demandeur, comme suit. Ajouter l'objet DIAG_RESPONSE à la fin de DREQ_in et mettre à jour la longueur du paquet. Si un objet ROUTE est présent dans le message, décrémenter le pointeur R et régler l'adresse de cible à la dernière adresse de l'objet ROUTE ; autrement régler l'adresse cible à celle du demandeur. Changer le champ Type dans l'en-tête commun de DREQ à DREP. Finalement, recalculer la somme de contrôle, envoyer la DREP à l'adresse cible, et revenir. Noter que le bit MF doit être à zéro dans ce cas.

Send_DREP : on entre dans cette séquence si le message DREQ augmenté du nouvel objet DIAG_RESPONSE est trop gros pour être transmis à l'expéditeur ou si il n'est pas éligible à la transmission, trop gros pour être retourné comme DREP.

SD1. Faire une copie de DREQ_in et changer le champ Type de message de DREQ en DREP. Couper tous les objets DIAG_RESPONSE de DREQ_in et ajuster le décalage de fragment. Le message DREP contient les objets DIAG_RESPONSE accumulés par les nœuds antérieurs.

SD2. Envoyer le message DREP vers le demandeur, comme suit. Si un objet ROUTE est présent dans le message DREP,

décrémenter le pointeur R et régler l'adresse de cible à la dernière adresse dans l'objet ROUTE ; autrement régler l'adresse cible à celle du demandeur. Établir le bit MF, recalculer la somme de contrôle et renvoyer le message DREP à l'adresse cible.

SD3. Si la taille réduite de DREQ_in plus la taille de DIAG_RESPONSE plus la taille d'une adresse IP (si il existe un objet ROUTE) est inférieure ou égale à la MTU du chemin, retourner alors à l'étape 8 de la séquence principale de traitement de DREQ ci-dessus.

SD4. Si il existe un objet ROUTE, le remplacer dans DREQ_in par un objet ROUTE vide et régler le bit d'erreur "Objet ROUTE trop gros" (0x04) dans la DIAG_RESPONSE. Dans l'un et l'autre cas, retourner à l'étape 8 de la séquence principale de traitement de DREQ ci-dessus.

4.2. Transmission de paquet DREP

Lorsque un objet ROUTE est présent, les messages DREP sont transmis bond par bond vers le demandeur, en inversant le chemin tel que mentionné dans l'objet ROUTE. Autrement, les messages DREP sont envoyés directement au demandeur original.

Lorsque un nœud reçoit un message DREP, il diminue simplement le pointeur R de un (longueur d'adresse) recalcule la somme de contrôle et transmet le message à l'adresse pointée par le pointeur R dans la liste de chemin. Si un nœud, autre que le dernier bond, reçoit un paquet DREP où le pointeur R est égal à zéro, il doit l'envoyer directement au demandeur.

Lorsque le nœud de dernier bond reçoit un message DREP, il envoie le message au demandeur.

4.3 Choix et ajustement de la MTU

Parce que le message DREQ porte la taille de MTU admise des bonds précédents que les messages DREP vont ensuite traverser, cette unique caractéristique permet une fragmentation sémantiquement facile comme décrit ci-dessus. Chaque fois que le message DREQ approche de la taille de la MTU du chemin, il peut être élagué avant une nouvelle transmission.

Lorsque un demandeur envoie un message DREQ, le champ MTU du chemin dans l'objet DIAGNOSTIC peut être réglé à une valeur par défaut configurée. Il est possible que la valeur originale de MTU du chemin soit choisie supérieure à la valeur de MTU réelle sur certaines portions du chemin tracé. Donc chaque nœud RSVP intermédiaire doit vérifier la valeur de MTU lorsque il traite un message DREQ. Si la valeur de MTU spécifiée est supérieure à la MTU de l'interface entrante (sur laquelle le message DREQ va être transmis) le nœud change la valeur de MTU dans l'en-tête pour une valeur plus petite.

Chaque fois que la taille d'un message DREQ devient supérieure à la valeur de la MTU du chemin, un nœud RSVP intermédiaire fait une copie du message, le convertit en un message DREP à renvoyer, et élague ensuite les résultats partiels du message DREQ. Si dans ce cas aussi la DREQ ne peut pas être transmise vers l'amont à cause d'un objet ROUTE trop gros, "Objet ROUTE trop gros" est établi et l'objet ROUTE est élagué. Par suite de l'élagage de l'objet ROUTE, les DREP vont aller bond par bond jusqu'à ce nœud et vont alors être immédiatement transmis à l'adresse du demandeur.

Même si les étapes montrées ci-dessus sont suivies, il y a quelques cas où la fragmentation va se produire à la couche IP. Par exemple, il peut exister des bonds non RSVP avec de plus petites MTU avant qu'on atteigne le dernier bond, ou si la réponse est renvoyée directement au demandeur (et non bond par bond) la DREP peut prendre un chemin différent vers le demandeur que celui de la DREQ depuis le demandeur. Un autre cas est lorsque il existe une liaison dont la MTU est plus petite que la valeur minimum de MTU de chemin définie au paragraphe 3.3.

4.4 Erreurs

Si une condition d'erreur empêche un message DREP d'être transmis plus loin, il est tout simplement éliminé.

Si une condition d'erreur, comme un manque d'état du chemin, empêche un message DREQ d'être transmis plus loin, le nœud doit changer le message actuel en type DREP et le retourner à l'adresse de réponse.

5. Diagnostic de problèmes en utilisant les facilités de diagnostic de RSVP

5.1 À travers des pare-feu

Les pare-feu peuvent causer des problèmes de transmission de message de diagnostic. Il y a deux cas différents.

D'abord, supposons que celui qui interroge réside sur un hôte receveur de la session à examiner. Dans ce cas, les pare-feu ne devraient pas empêcher la transmission des messages de diagnostic bond par bond, en supposant que les trous appropriés ont été percés dans le pare-feu pour permettre la transmission bond par bond pour les autres messages RSVP. Celui qui interroge peut commencer en n'incluant pas d'objet ROUTE, ce qui peut donner une livraison de réponse plus rapide et réduire les frais généraux aux nœuds intermédiaires. Cependant, si aucune réponse n'est reçue, l'interrogateur peut envoyer à nouveau le message DREQ avec un objet ROUTE, en spécifiant qu'une réponse bond par bond devrait être envoyée.

Si le demandeur est un hôte tiers qui est séparé de l'adresse de dernier bond par un pare-feu (soit le demandeur est derrière un pare-feu, soit le dernier bond est un nœud derrière un pare-feu, soit les deux) on ne connaît pas pour le moment d'autre solution que de changer le dernier bond en un nœud qui soit sur le même côté du pare-feu que le demandeur.

5.2 Examen des temporisateurs RSVP

On peut facilement collecter des information sur la valeur actuelle du temporisateur à chaque bond RSVP le long du chemin. Cela sera très utile dans les situations où l'état de réservation change très fréquemment, pour découvrir si les changements d'état sont dus à un réglage inapproprié des valeurs du temporisateur, ou des valeur de K (quand on traverse des liaisons à pertes) ou de fréquents changements d'acheminement.

5.3 Découverte de nuages non RSVP

Le champ D-TTL dans chaque objet DIAG_RESPONSE montre le nombre de bonds d'acheminement entre les nœuds adjacents RSVP. Donc, toute valeur supérieure à un indique qu'on traverse un nuage non RSVP. Conjointement aux horodatages d'arrivée (en supposant que NTP fonctionne) cette valeur peut aussi donner une indication vague, mais pas nécessairement précise, de la grosseur de ce nuage. On peut aussi découvrir tous les nœuds intermédiaires non RSVP en faisant passer des traceroute en envoi individuel ou en diffusion groupée.

5.4 Découverte de fusions de réservation

La valeur de spécification de flux (*flowspec*) dans un objet DIAG_RESPONSE spécifie la quantité de ressources qui est réservée pour le flux de données défini par la spécification de filtre dans le même bloc de données. Lorsque cette valeur d'objets DIAG_RESPONSE adjacents diffère, c'est-à-dire, lorsque un nœud amont Rd a une valeur inférieure à celle du nœud Ru immédiatement en amont, cela indique une fusion de réservation avec la ou les demandes RSVP provenant d'autres interfaces vers l'aval en Rd. De plus, en cas de réservation de style partagé explicite (SE, *Shared Explicit*) on peut examiner comment les différentes portées de SE sont fusionnées à chaque bond.

En particulier, si un receveur envoie un message DREQ avant d'envoyer sa propre réservation, il peut découvrir (1) combien de bonds RSVP sont le long du chemin entre l'envoyeur spécifié et lui-même, (2) combien de bonds ont déjà des réservations par d'autres receveurs, et (3) éventuellement une précision grossière de si sa demande de réservation pourrait être fusionnées à d'autres existantes.

5.5 Diagnostics d'erreur

En plus d'examiner l'état des réservations qui fonctionnent, les messages de diagnostic RSVP ne seront probablement pas invoqués lorsque les choses ne fonctionnent pas correctement. Par exemple, un receveur a réservé un tuyau adéquat pour un flux spécifié de données entrantes, mais le retard ou taux de perte observé est bien plus élevé qu'attendu. Dans ce cas, le receveur peut utiliser la facilité de diagnostic pour examiner l'état de réservation à chaque bond RSVP le long du chemin pour trouver si l'état RSVP est réglé correctement, si il y a un trou noir sur le chemin et qui a causé des pertes de messages RSVP, ou si il y a des nuages non RSVP, et où ils sont, qui peuvent avoir causé le problème de performance.

5.6 Traversée de routeurs RSVP "traditionnels"

Comme cette facilité de diagnostics a été développée et ajoutée à RSVP après la mise en place d'un certain nombre de mises en œuvre de RSVP, il est possible, ou même probable, que lorsque on effectue un diagnostic RSVP, on puisse rencontrer un ou plusieurs nœuds à capacité RSVP qui ne comprennent pas les messages de diagnostic et les éliminent.

Lorsque cela se produit, le client qui l'invoque n'aura pas de réponse à ses demandes.

On façon de contourner ces nœuds RSVP "traditionnels" est d'effectuer de façon répétée le diagnostic RSVP, guidé par les informations de traceroute, ou mtrace en cas de diffusion groupée. Lorsque une interrogation de diagnostic RSVP arrive en fin de temporisation (voir au paragraphe suivant) on peut d'abord utiliser traceroute pour obtenir la liste des nœuds le long du chemin, et ensuite augmenter graduellement la valeur du champ Max-bonds-RSVP dans le message DREQ, en commençant par une valeur faible jusqu'à ce qu'on ne reçoive plus de réponse. On peut alors essayer à nouveau le diagnostic RSVP en commençant par le premier nœud (qui est plus en amont vers l'expéditeur) après celui qui ne répond pas.

Il y a deux problèmes avec la méthode mentionnée ci-dessus dans le cas de sessions en envoi individuel. Les deux problèmes sont reliés au fait que les informations de traceroute donnent le chemin depuis le demandeur jusqu'à l'expéditeur. Le premier problème est que le dernier bond peut n'être pas sur le chemin qui va du demandeur à l'expéditeur. Dans ce cas, on ne peut obtenir les informations que de la portion de chemin qui va du dernier bond à l'expéditeur qui coupe le chemin qui va du demandeur à l'expéditeur. Si les routeurs qui ne sont pas sur l'intersection des deux chemins n'ont pas d'état du chemin pour la session diagnostiquée, ils vont alors répondre par R-erreur=0x01. Le demandeur peut surmonter ce problème en envoyant une DREQ à chaque routeur que le chemin (de lui-même à l'expéditeur) jusqu'à ce qu'il atteigne le premier routeur qui appartient au chemin de l'expéditeur au dernier bond.

Le second problème est que traceroute donne le chemin du demandeur à l'expéditeur qui, du fait des asymétries d'acheminement, peut être différent du chemin qu'utilise le trafic de l'expéditeur au dernier bond. Il y a (au moins) un cas où cette asymétrie va causer l'échec du diagnostic. On le présente ci-dessous.

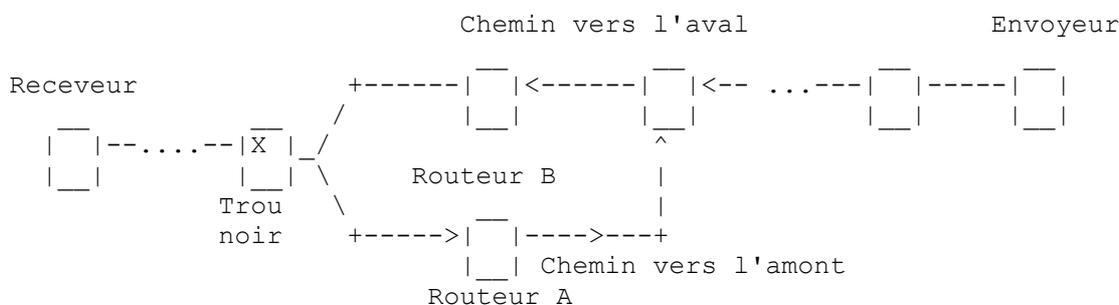


Figure 2

Ici le premier bond en amont du trou noir est différent sur le chemin vers l'amont et sur le chemin vers l'aval. Traceroute va indiquer le routeur A comme bon précédent (au lieu du routeur B qui est le bon). Envoyer une DREQ au routeur A va avoir pour résultat que A va répondre avec R-erreur 0x01 (Pas d'état du chemin). Si les deux chemins convergent encore, le demandeur peut utiliser la solution proposée plus haut pour obtenir des informations (partielles) de la part du reste du chemin.

On n'a pas, pour le moment, de solution complète pour les scénarios problématiques décrits ici.

6. Commentaires sur la mise en œuvre de client de diagnostic

Suivant le principe de conception que les nœuds dans le réseau ne devraient pas conserver plus d'état que nécessaire, les nœuds RSVP ne sont chargés que de la transmission des messages Diagnostic et de remplir les objets DIAG_RESPONSE. Des fonctions de diagnostic supplémentaires ne devraient être assumées que par les clients de diagnostic. De plus, si la fonction de diagnostic est invoquée à partir d'un hôte tiers, on ne devrait pas exiger que cet hôte fasse fonctionner un robot RSVP pour effectuer cette fonction. On décrit ci après les fonctions de base qu'un robot de client de diagnostic devrait effectuer.

1. Prendre les entrées de l'utilisateur sur la session à diagnostiquer, le dernier bond et l'adresse de l'expéditeur, le Max-bonds-RSVP, et éventuellement la liste DIAG_SELECT ; créer un message DREQ et l'envoyer au nœud RSVP de dernier bond en utilisant un message IP brut avec le numéro de protocole 46 (RSVP). Si l'utilisateur a spécifié que la réponse devrait être envoyée bond par bond, inclure un objet ROUTE vide dans le message DREQ envoyé. Régler la MTU de chemin à la plus petite de la demande de l'utilisateur et de la MTU de la liaison à travers laquelle la DREQ sera envoyée. L'accès de la prise UDP sur laquelle écoute le client de diagnostic pour ses réponses devrait être inclus dans l'objet FILTER_SPEC du demandeur.

2. Établir un temporisateur de retransmission, en attente de la réponse (un ou plusieurs messages DREP). Écouter sur l'accès UDP spécifié les réponses provenant du nœud RSVP de dernier bond. Le nœud RSVP de dernier bond, à réception de messages DREP, les envoie au client de diagnostic comme paquets UDP, en utilisant l'accès fourni dans l'objet FILTER_SPEC du demandeur.
3. À réception d'un message DREP sur une demande de diagnostic en cours, le client devrait remettre à zéro le temporisateur de retransmission, vérifier si la réponse contient le résultat complet du diagnostic demandé. Si il en est ainsi, il devrait passer immédiatement le résultat à l'entité invoquante.
4. Réassembler les fragment de DREP. Si la première réponse à une demande de diagnostic en cours ne contient qu'un fragment du résultat attendu, le client devrait lancer un temporisateur de réassemblage d'une façon similaire au temporisateur de réassemblage de paquet IP. Si le temporisateur arrive à expiration avant que tous les fragments soient arrivés, le client devrait passer le résultat partiel à l'entité invoquante.
5. Utiliser les temporisateurs de retransmission et de réassemblage pour traiter en douceur les pertes de paquet et les scénarios de réponse de fragment. En l'absence de réponse à la première demande de diagnostic, un client devrait retransmettre la demande quelques fois. Si toutes les retransmissions échouent aussi, le client devrait invoquer traceroute ou mtrace pour obtenir la liste de bonds le long du segment de chemin à diagnostiquer, et ensuite effectuer une itération de diagnostics avec une augmentation du compte de bonds, comme suggéré au paragraphe 5.6 afin de traverser les nœuds à capacité RSVP sans capacité de diagnostic.
6. Si tous les efforts ci-dessus échouent, le client doit le notifier à l'entité invoquante.

7. Considérations sur la sécurité

Les diagnostics RSVP, comme tous les autres outils de diagnostic, peuvent présenter un risque pur la sécurité car ils peuvent révéler des informations d'état de RSVP éventuellement sensibles à des tiers non habitués à les connaître.

On estime que ce risque est minime, car comme expliqué dans l'introduction les messages de diagnostics ne produisent pas d'effet collatéral et ne peuvent donc pas changer l'état RSVP dans les nœuds. À cet égard, les diagnostics RSVP sont moins un menace pour la sécurité que d'autres outils et protocoles de diagnostic tels que SNMP.

De plus, le traitement des messages Diagnostic peut être désactivé si on estime qu'il y a une menace pour la sécurité.

8. Remerciements

L'idée de développer une facilité de diagnostic pour RSVP a été d'abord suggérée par Mark Handley de ACIRI. Nos remerciements s'adressent à Lee Breslau de AT&T Labs et à John Krawczyk de Nortel Networks pour leurs précieux commentaires sur le premier jet de ce mémoire. Lee Breslau, Bob Braden, et John Krawczyk ont contribué par d'autres commentaires après la réunion de mars 1996 de l'IETF. Steven Berson a fourni des commentaires précieux sur divers projets de ce mémoire. Tim Gleeson a contribué à une impressionnante liste de commentaires rédactionnels. On tient aussi à remercier Intel pour son allocation de recherche qui a contribué au soutien de ce travail. Subramaniam Vincent a fait la plus grande partie de ce travail comme assistant de recherche en doctorat à l'Information Sciences Institute (ISI) de l'USC.

9. Références

- [RFC2205] R. Braden, éd., et autres, "[Protocole de réservation de ressource \(RSVP\)](#) -- version 1, spécification fonctionnelle", septembre 1997. (*MàJ par RFC2750, RFC3936, RFC4495, RFC6780*) (P.S.)
- [RFC2746] A. Terzis, J. Krawczyk, J. Wroclawski, L. Zhang, "Fonctionnement de [RSVP sur tunnels IP](#)", janvier 2000. (P.S.)

10. Adresse des auteurs

Andreas Terzis
UCLA
4677 Boelter Hall
Los Angeles, CA 90095
téléphone : 310-267-2190
mél : terzis@cs.ucla.edu

Subramaniam Vincent
Cisco Systems
275, E Tasman Drive, MS SJC04/2/1
San Jose, CA 95134
téléphone : 408 525 3474
mél : svincent@cisco.com

Bob Braden
USC Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292
téléphone : 310 822-1511
mél : braden@isi.edu

Lixia Zhang
UCLA
4531G Boelter Hall
Los Angeles, CA 90095
téléphone : 310-825-2695
mél : lixia@cs.ucla.edu

10. Déclaration complète de droits de reproduction

Copyright (C) The Internet Society (2000). Tous droits réservés.

Ce document et les traductions de celui-ci peuvent être copiés et diffusés, et les travaux dérivés qui commentent ou expliquent autrement ou aident à sa mise en œuvre peuvent être préparés, copiés, publiés et distribués, partiellement ou en totalité, sans restriction d'aucune sorte, à condition que l'avis de droits de reproduction ci-dessus et ce paragraphe soient inclus sur toutes ces copies et œuvres dérivées. Toutefois, ce document lui-même ne peut être modifié en aucune façon, par exemple en supprimant le droit d'auteur ou les références à l'Internet Society ou d'autres organisations Internet, sauf si c'est nécessaire à l'élaboration des normes Internet, auquel cas les procédures pour les droits de reproduction définis dans les processus des normes pour l'Internet doivent être suivies, ou si nécessaire pour le traduire dans des langues autres que l'anglais.

Les permissions limitées accordées ci-dessus sont perpétuelles et ne seront pas révoquées par la Société Internet, ses successeurs ou ayants droit.

Ce document et les renseignements qu'il contient sont fournis "TELS QUELS" et l'INTERNET SOCIETY et l'INTERNET ENGINEERING TASK FORCE déclinent toute garantie, expresse ou implicite, y compris mais sans s'y limiter, toute garantie que l'utilisation de l'information ici présente n'enfreindra aucun droit ou aucune garantie implicite de commercialisation ou d'adaptation à un objet particulier.

Remerciement

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