Groupe de travail Réseau
Request for Comments : 5401
RFC rendue obsolète : 3941
Catégorie : Sur la voie de la normalisation
Traduction Claude Brière de L'Isle

B. Adamson, Naval Research Laboratory
C. Bormann, Universitaet Bremen TZI
M. Handley, University College London
J. Macker, Naval Research Laboratory
novembre 2008

Blocs de construction d'accusé de réception négatif (NACK) de diffusion groupée

Statut du présent mémoire

Le présent document spécifie un protocole de l'Internet sur la voie de la 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.

Résumé

Le présent document discute de la création de protocoles fiables de diffusion groupée qui utilisent des retours d'accusé de réception négatifs (NACK, *negative-acknowledgment*). Il présente les raisons des buts de conception du protocole et ses hypothèses. Il identifie les défis techniques du fonctionnement de protocole de diffusion groupée fiable fondé sur le NACK (et dans certains cas généraux). Ces buts et ces défis se résolvent en un ensemble de "blocs de construction" fonctionnels qui visent différents aspects du fonctionnement de protocole de diffusion groupée fiable. Il est prévu que ces blocs de construction soient utiles pour générer différentes instanciations de protocoles de diffusion groupée fiable. Le présent document rend obsolète la RFC 3941.

Table des matières

1. Introduction

Le transport fiable de diffusion groupée est une technologie désirable pour une distribution efficace et fiable de données à un groupe sur l'Internet. Les complexités des paradigmes de la communication de groupe nécessitent différents types et instances de protocoles pour satisfaire la gamme d'exigences de performances et d'adaptabilité des différentes applications et utilisateurs potentiels de diffusion groupée fiable (voir la [RFC2357]). Le présent document traite de la création de protocoles de diffusion groupée fiable qui utilisent des retours d'accusés de réception négatifs (NACK, *Negative Acknowledgment*). Les protocoles fondés sur le NACK entraînent généralement des messages de retour moins fréquents que les protocoles de fiabilité fondés sur l'accusé de réception positif (ACK, *acknowledgment*). Ces messages moins fréquents de retour simplifient le problème de l'explosion de retours lorsque la taille de groupe augmente. Bien que différentes instances de protocoles puissent être exigées pour satisfaire les demandes spécifiques d'application et d'architecture de réseau [ArchConsiderations], il y a un certain nombre de composants fondamentaux qui peuvent être communs à ces différentes instanciations.

Le présent document décrit le cadre et les composants communs de "bloc de construction" pertinents pour les protocoles de diffusion groupée qui sont fondés principalement sur le fonctionnement de NACK pour un transport fiable. Bien que ce document discute un large ensemble de composants de diffusion groupée fiable et de problèmes pertinents pour la diffusion groupée fiable fondée sur le concept de protocole de NACK, il traite spécifiquement le détail des blocs de construction suivants, qui ne sont pas traités dans d'autres documents de l'IETF :

1. les stratégies de transmission d'envoyeur de diffusion groupée fondée sur le NACK ,
2. le processus de réparation de NACK avec suppression de retour fondé sur un temporisateur, et
3. le temps d'aller-retour pour adapter le NACK et les autres temporisateurs.

Les mises en œuvre de diffusion groupée fiable fondée sur le NACK DEVRAIENT utiliser les techniques de codage d'écrasement de correction d'erreur directe (FEC, *Forward Error Correction*) décrites dans le document sur les blocs de construction de FEC [RFC5052]. Le codage de l'écrasement au niveau du paquet permet que les paquets manquants d'un certain bloc de FEC soient récupérés en utilisant les paquets de parité au lieu de la retransmission classique individualisée du contenu des données de source originales. Pour cette raison, le présent document se réfère aux mécanismes de protocole pour la fiabilité comme à un "processus de réparation". Noter que les protocoles fondés sur le NACK peuvent fournir de façon réactive les paquets de parité en réponse aux demandes du receveur pour une réparation plutôt que de juste envoyer de façon proactive le contenu ajouté de parité de FEC au titre de la transmission originale. Une utilisation hybride proactive/réactive du contenu de FEC est aussi possible avec les mécanismes décrits dans ce document. Certaines classes de codage de FEC, comme les codes de distance maximale séparable (MDS, *Maximal Separable Distance*) permettent aux envoyeurs de mettre en œuvre de façon dynamique des stratégies déterministes, très efficaces, de réparation de groupe de receveurs au titre du schéma de répétition automatique de demande (ARQ, *Automated Repeat-reQuest*) sélective fondé sur le NACK.

Les relations potentielles aux autres blocs de construction de transport de diffusion groupée fiable (par exemple, FEC, contrôle d'encombrement) et les questions générales avec les protocoles de diffusion groupée fiable fondée sur le NACK sont aussi discutées. Le présent document suit les lignes directrices de la [RFC3269].

Déclaration d'intention
Le présent mémoire contient des descriptions de blocs de construction qui peuvent être appliqués dans la conception de protocoles de diffusion groupée fiable qui utilisent des retours d'accusés de réception négatifs (NACK). La [RFC3941] contient une description précédente de cette spécification. La RFC 3941 a été publiée dans la catégorie "Expérimentale". Il était l'intention déclarée du groupe de travail Transport de diffusion groupée fiable (RMT, *Reliable Multicast Transport*) de reproposer cette spécification comme proposition de norme de l'IETF en temps utile.

Cette spécification de proposition de norme se fonde donc sur la [RFC3941] et a été mise à jour en accord avec l'expérience accumulée et la maturité croissante du protocole depuis la publication de la RFC 3941. Cette expérience s'applique à cette spécification elle-même et aux stratégies de contrôle d'encombrement relatives à l'utilisation de cette spécification. Les différences entre la [RFC3941] et ce document sont décrites à la Section 6.

1.1 Langage des exigences

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

2. Motifs

Chaque instance potentielle de protocole utilisant les blocs de construction présentés ici (et dans les autres documents applicables aux blocs de construction) va avoir des critères spécifiques qui peuvent influencer la conception d'un protocole individuel. Pour prendre en charge le développement des blocs de construction applicables, il est utile d'identifier et résumer les buts et hypothèses qui commandent la conception générale du protocole. Ce sont des zones que chaque instance de protocole va devoir traiter en détail. Chaque description de bloc de construction de ce document inclura une discussion de l'impact de ces critères de conception. Les catégories de critères de conception considérées ici incluent :

1. le modèle de service de livraison,
2. la dynamique d'adhésion au groupe,
3. les relations envoyeur/receveur,
4. l'adaptabilité de la taille de groupe,
5. les performances de livraison des données, et
6. les environnements de réseau.

Toutes ces zones sont discutées au moins brièvement. De plus, d'autres documents de transport de bloc de construction de diffusion groupée fiable, comme la [RFC5052], ont été créés pour traiter des zones qui sortent du domaine d'application du présent document. Les instances de protocole de diffusion groupée fiable fondée sur le NACK peuvent dépendre de ces autres blocs de construction autant que de ceux présentés ici. Le présent document se concentre sur les zones qui sont uniques à la diffusion groupée fiable fondée sur le NACK mais qui peuvent être utilisées de concert avec les autres zones de blocs de construction. Dans certains cas, un bloc de construction peut être capable de traiter une large gamme d'hypothèses, tandis que dans d'autres cas, un compromis va être exigé pour satisfaire les différents besoins des applications ou environnements de fonctionnement. Lorsque nécessaire, les caractéristiques de bloc de construction sont désignées comme des paramètres pour satisfaire différentes exigences. Bien sûr, un but sous-jacent va être de minimiser la complexité de conception et d'au moins recommander les valeurs par défaut pour tous ces paramètres qui satisfont une exigence générale de "transfert de données en vrac" dans un environnement Internet normal. Les formes de "transfert de données en vrac" couvertes ici incluent le transport fiable de contenu en vrac, de longueur fixe, à priori statique et aussi la transmission de contenu non prédéterminé, peut-être en flux, de longueur indéfinie. Le paragraphe 3.5 discute plus en détail de ces différentes formes de contenus de données en vrac.

2.1 Modèle de service de livraison

Le but implicite d'un protocole de transport de diffusion groupée fiable est la livraison fiable des données parmi un groupe de membres qui communiquent en utilisant le service de datagrammes de diffusion groupée IP. Cependant, le service spécifique que l'application tente de fournir peut impacter les décisions de conception. Le modèle de service le plus basique pour le transport de diffusion groupée fiable est celui du "transfert en vrac", qui est un point principal du présent document et des autres documents du groupe de travail RMT en rapport. Cependant, les mêmes principes de conception de protocole peuvent aussi être appliqués aux autres modèles de service, par exemple, des échanges plus interactifs de petits messages comme avec des discussions sur un tableau blanc ou de parlotte. Au sein de ces différents modèles , il y a des problèmes comme la capacité de l'envoyeur à mettre en antémémoire les données transmises (ou l'état qui y fait référence) pour retransmission ou réparation. Les besoins d'ordre et/ou de causalité dans la séquence des transmissions et réceptions parmi les membres du groupe peuvent être différents selon le contenu des données. Le paradigme de la communication de groupe diffère significativement du modèle point à point en ce que, selon le type de contenu de données, certains receveurs peuvent achever la réception d'une portion du contenu de données et être capables d'agir sur lui avant que d'autres membres aient reçu le contenu. Cela peut être acceptable (ou même désirable) pour certaines applications mais pas pour d'autres. Ces exigences variables amènent au besoin d'un certain nombre de différentes conceptions d'instanciations de protocole. Un défi significatif du développement de mécanismes généralement utiles de blocs de construction s'accommode de même une gamme limitée de ces capacités sans définir de détails spécifiques au niveau de l'application.

Un autre facteur qui impacte le modèle de service de livraison est la possibilité que différents receveurs dans le groupe de diffusion groupée aient une qualité significativement différente de connexité au réseau. Cela peut impliquer des receveurs

avec un débit très limité à cause du débit de connexion ou de pertes de paquet substantielles. Les mises en œuvre de protocole fondé sur le NACK peuvent souhaiter fournir des politiques par lesquelles des receveurs avec des performances extrêmement faibles sont exclus du groupe principal ou évacués sur un groupe de livraison séparé. Noter que certains modèles d'application peuvent exiger que le groupe entier soit contraint aux performances du "membre le plus faible" pour satisfaire des exigences de fonctionnement. Dans l'un et l'autre cas, la conception du protocole devrait considérer cet aspect du modèle de service de livraison de diffusion groupée fiable.

2.2 Dynamique de l'adhésion de groupe

Un domaine dans lequel la communication de groupe peut différer des communications en point à point est que même si la composition du groupe change, le "fil" de communication peut encore exister. Cela contraste avec le modèle de communication en point à point où, si l'une des deux parties s'en va, le processus de communication (échange de données) cesse (ou au moins fait une pause). Selon les buts de l'application, les envoyeurs et receveurs participant à une "session" de transport de diffusion groupée fiable peuvent être capables de se joindre plus tard, quitter et/ou potentiellement rejoindre alors que le "fil" de la communication de groupe en cours reste encore fonctionnel et utile. Noter aussi que cela peut impacter le contenu du message de protocole. Si les "entrants tardifs" sont acceptés, une certaine quantité d'informations supplémentaires peut être placée dans les en-têtes de message pour s'accommoder de cette fonctionnalité. Autrement, les informations peuvent être envoyées dans un message propre (sur demande ou de façon intermittente) si l'impact de frais généraux des transmissions de message normales est estimé trop grand. Les dynamiques de groupe peuvent aussi impacter d'autres mécanismes de protocole comme le rythme des NACK, le fonctionnement du contrôle d'encombrement, etc.

2.3 Relations envoyeur/receveur

Les relations des envoyeurs et des receveurs parmi les membres du groupe doivent être examinées. Dans certaines applications, il peut y avoir un seul envoyeur qui diffuse à un groupe de receveurs. Dans d'autres cas, il peut y avoir plus d'un envoyeur ou la possibilité que chacun dans le groupe soit un envoyeur et un receveur des données .

2.4 Adaptabilité de la taille de groupe

La diffusion groupée IP native [RFC1112] peut s'adapter à des tailles de groupe extrêmement grandes. Il peut être désirable pour certaines applications de s'adapter à la capacité de l'infrastructure de diffusion groupée. Dans sa forme la plus simple, il y a des limites à l'estimation de la taille de groupe à laquelle un protocole fondé sur le NACK peut être appliqué sans que le volume potentiel de messages de NACK en retour surpasse la capacité du réseau. On appelle souvent cela une "explosion de retours". Les recherches suggèrent que des tailles de groupe de diffusion groupée fiable fondée sur le NACK de l'ordre de dizaines de milliers de receveurs peuvent opérer avec des niveaux acceptables de retours à l'envoyeur en utilisant des techniques de suppression probabilistes, fondées sur le temporisateur [NormFeedback]. Plutôt que les receveurs transmettent immédiatement les messages de rétroaction quand une perte est détectée, ces techniques spécifient l'utilisation de temporisations à retard aléatoire intentionnellement adapté de telle façon que certains receveurs potentiels qui envoient des NACK puissent supprimer d'eux-même leurs retours quand ils entendent les messages provenant des autres receveurs qui ont choisi des intervalles plus courts de temporisation aléatoire de retours. Cependant, il peut y avoir des heuristiques supplémentaires de suppression de NACK appliquées pour permettre à ces protocoles de s'adapter à des tailles de groupe encore plus grandes. Dans le cas de grandes tailles, il peut être prohibitif pour les membres de maintenir l'état de tous les autres membres (en particulier, des autres receveurs) du groupe. L'impact de l'estimation de la taille de groupe doit être pris en compte dans le développement des blocs de construction applicables.

L'adaptabilité de l'estimation de la taille de groupe peut aussi être aidée par l'assistance de système intermédiaire ; voir le paragraphe 2.7.

2.5 Performances de la livraison des données

Il y a un compromis entre l'adaptabilité et la latence de livraison des données quand on conçoit des protocoles fondés sur le NACK. Si la suppression de NACK probabiliste fondée sur le temporisateur est utilisée, il va y avoir des délais incorporés dans le processus de NACK pour permettre que la suppression se fasse et pour permettre à l'envoyeur des données d'identifier le contenu approprié pour une transmission de réparation efficace. Par exemple, les temporisations de retard peuvent être utilisées pour assurer une suppression efficace de NACK et la transmission de réparation, mais cela viendra au prix d'une latence de livraison accrue et d'exigences de mise en mémoire tampon accrues pour les envoyeurs et les receveurs. Les blocs de construction DEVRAIENT permettre aux applications d'établir des limites pour les performances de livraison des données. Noter que les concepteurs d'applications doivent connaître le compromis d'adaptabilité qui est

fait quand de telles limites sont appliquées.

2.6 Environnements de réseau

Le protocole Internet a historiquement assumé le rôle de fournisseur de service à travers des topologies de réseau hétérogènes. Il est désirable qu'un protocole de diffusion groupée fiable soit capable d'opérer efficacement à travers une large gamme de réseaux auxquels s'applique le service IP général. La bande passante disponible sur les liaisons entre les membres d'un seul groupe peut varier aujourd'hui entre un faible nombre de kbit/s pour les liaisons sans fil et plusieurs Gbit/s pour des connexions de LAN à haut débit, avec des degrés variés de contention provenant des autres flux. Récemment, un certain nombre de services réseau asymétriques incluant des modems 56k/ADSL, le service de TV Internet par câble, de satellite, et autres services de communication sans fil, ont commencé à proliférer. Beaucoup d'entre eux sont des supports de diffusion par nature avec un éventail potentiellement grand auquel le service de diffusion groupée IP est directement applicable. De plus, des questions de politique et/ou techniques peuvent résulter en des topologies où la connexité de diffusion groupée est limitée au modèle de diffusion groupée spécifique de source (SSM, *Source-Specific Multicast*) provenant d'une source spécifique [RFC4607]. Les receveurs dans le groupe peuvent être restreints à des retours en envoi individuel pour les NACK et autres messages. La conception du protocole et le développement de bloc de construction doit prendre en compte la nature des réseaux sous-jacents.

2.7 Assistance de système intermédiaire

L'assistance intermédiaire des appareils/systèmes qui ont une connaissance directe de la topologie de réseau sous-jacente peut être utilisée pour augmenter les performances et l'adaptabilité des protocoles de diffusion groupée fiable fondée sur le NACK. L'agrégation et le filtrage des rétroactions des données de réparation de l'envoyeur peuvent être possibles avec des protocoles fondés sur le NACK utilisant des stratégies de réparation fondées sur la FEC, comme décrit dans le présent document et les autres documents de bloc de construction de transport de diffusion groupée fiable. Cependant, il va continuer d'y avoir un certain nombre d'instances où l'assistance d'un système intermédiaire n'est pas disponible ou praticable. Tous les composants de bloc de construction pour la diffusion groupée fiable en mode NACK DEVRONT être capables de fonctionner sans une telle assistance. Cependant, il est RECOMMANDÉ que ces protocoles envisagent aussi d'utiliser ces caractéristiques quand elles sont disponibles.

3. Fonctionnalité

La section précédente a présenté le rôle des blocs de construction de protocole et certains des critères qui peuvent affecter l'identification/conception de bloc de construction de diffusion groupée fiable fondée sur le NACK. La présente section décrit les différentes zones de bloc de construction applicables aux protocoles de diffusion groupée fiable fondée sur le NACK. Certaines de ces zones sont spécifiques des protocoles fondés sur le NACK. Les descriptions détaillées de ces zones sont fournies. Dans d'autres cas, les zones (par exemple, identifiants de nœuds, correction d'erreur directe (FEC) etc.) peuvent être applicables à d'autres formes de diffusion groupée fiable. Dans ces cas, la discussion ci-dessous décrit les exigences placées sur ces zones générales de bloc de construction du point de vue de la diffusion groupée fiable fondée sur le NACK. Lorsque applicable, les autres documents de bloc de construction sont référencés pour une possible contribution aux protocoles de diffusion groupée fiable fondée sur le NACK.

Pour chaque bloc de construction, une notion de "description d'interface" est fournie pour illustrer les dépendances d'un composant de bloc de construction à un autre composant ou à d'autres paramètres de protocole. Un composant de bloc de construction peut exiger certaine forme "d'entrée" d'un autre composant de bloc de construction ou d'une autre source pour exécuter sa fonction. Toute"entrée" exigée par un composant de bloc de construction et/ou tout "résultat" qui en découle va être défini et décrit dans chaque description d'interface de composant de bloc de construction. Noter que l'ensemble des blocs de construction présentés ici ne satisfait pas pleinement les besoins "d'entrée" et de "sortie" de chaque autre. Dans certains cas, les "entrées" pour les blocs de construction doivent venir d'autres blocs de construction externes au présent document (par exemple, le contrôle d'encombrement ou la FEC). Dans d'autres cas de bloc de construction de diffusion groupée fiable fondée sur le NACK, les "entrées" doivent être satisfaites par l'instanciation ou mise en œuvre du protocole spécifique (par exemple, données d'application et contrôle).

Les composants de bloc de construction suivants, pertinents pour la diffusion groupée fiable fondée sur le NACK, sont identifiés :

Spécifiques de la diffusion groupée fiable en mode NACK (NORM, *NACK-Oriented Reliable Multicast*)
1. Transmission d'envoyeur de diffusion groupée

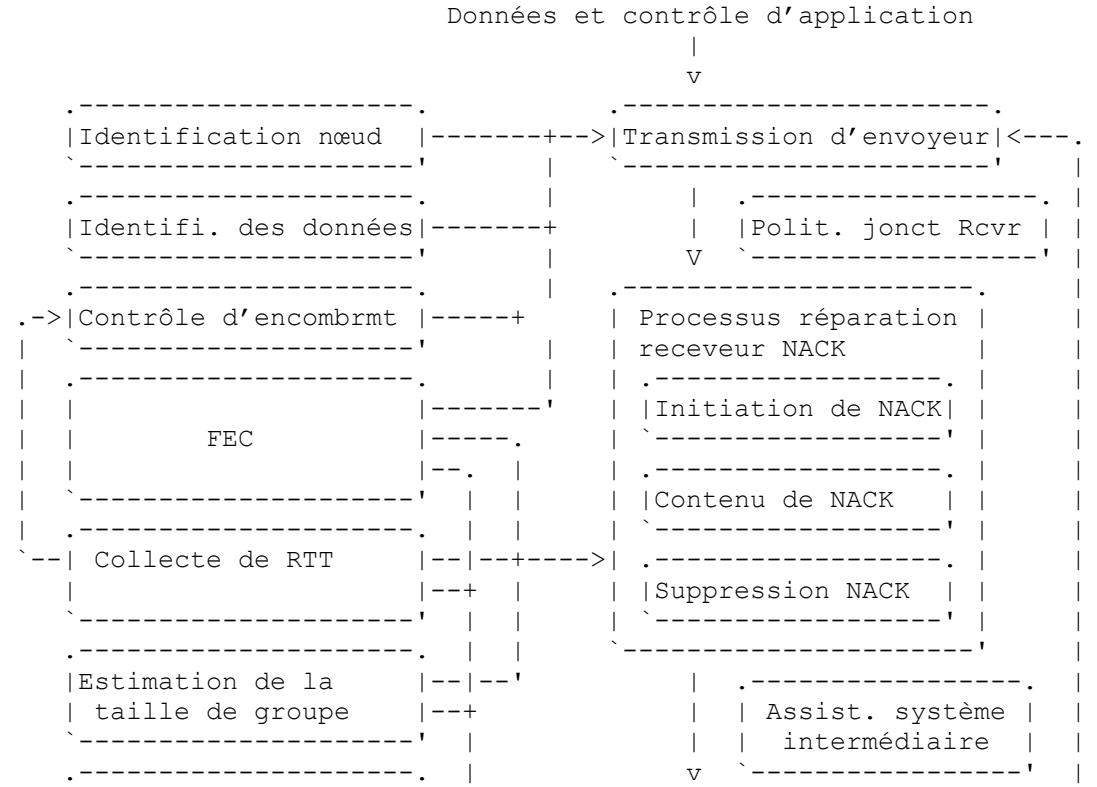
2. Processus de réparation de NACK
3. Politiques et procédures de jonction de receveur de diffusion groupée

D'objet général :
1. Identification de nœud (membre)
2. Identification du contenu des données
3. Correction d'erreur directe (FEC)
4. Collecte des temps d'aller-retour
5. Détermination/estimation de la taille de groupe
6. Fonctionnement du contrôle d'encombrement
7. Assistance de système intermédiaire
8. Mécanismes de protocole auxilliaire

La Figure 1 donne une vue d'ensemble graphique de ces zones de bloc de construction et de certaines de leurs relations. Par exemple, le contenu des messages de données qu'un envoyeur transmet initialement dépend des composants "Identification de nœud", "Identification du contenu des données", et "FEC", tandis que le taux de transmission de message va généralement dépendre du composant "Contrôle d'encombrement". Par conséquent, la réponse du receveur à ces transmissions (par exemple, des NACK pour la réparation) va dépendre du contenu de message de données et des entrées provenant d'autres composants de bloc de construction. Finalement, le traitement de l'envoyeur des réponses de receveur va rétroagir dans sa stratégie de transmission.

Les composants du côté gauche de cette figure sont les zones qui peuvent être applicables au delà de la diffusion groupée fiable fondée sur le NACK. Les plus significatifs de ces composants sont discutés dans d'autres documents de bloc de construction, comme le bloc de construction FEC [RFC5052]. De brèves descriptions de ces zones et leurs rôles dans les protocoles de diffusion groupée fiable fondée sur le NACK sont donnés ci-dessous, et la "Collecte de RTT" est discutée en détails au paragraphe 3.7 de ce document.

Les composants du côté droit sont vus comme spécifiques des protocoles de diffusion groupée fiable fondée sur le NACK, dont le plus notable est le processus de réparation de NACK. Ces zones sont discutées en détails ci-dessous (notamment, "Transmission d'envoyeur de diffusion groupée" et "Processus de réparation de NACK" aux paragraphes 3.1 et 3.2). Certains autres composants (par exemple, "Sécurité") impactent de nombreux aspects du protocole, et d'autres peuvent être plus transparents au cœur du traitement du protocole. Lorsque applicable, des recommandations techniques spécifiques sont données pour les mécanismes qui vont correctement satisfaire les buts du transport de diffusion groupée fiable fondée sur le NACK pour l'Internet.



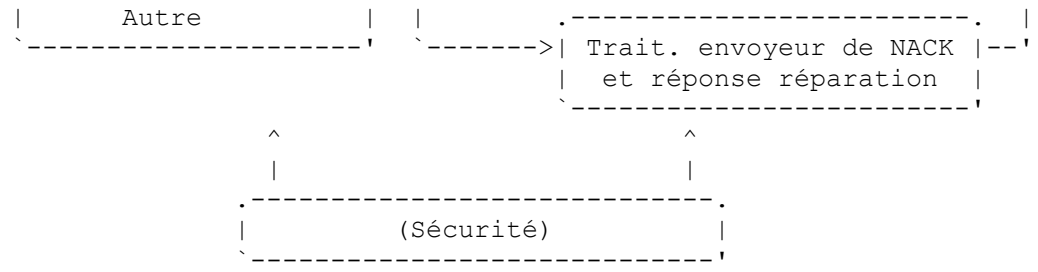


Figure 1 : Cadre de bloc de construction de diffusion groupée fiable fondée sur le NACK

3.1 Transmission d'envoyeur de diffusion groupée

Les envoyeurs de diffusion groupée fiable vont transmettre le contenu de données à la session de diffusion groupée. Le contenu de données va être dépendre de l'application. L'envoyeur va transmettre le contenu de données à un taux, et avec des tailles de message, déterminés par l'application et/ou les exigences de l'architecture du réseau. Tout codage de FEC des transmissions de l'envoyeur DEVRAIT se conformer aux lignes directrices des blocs de construction de FEC [RFC5052]. Quand des mécanismes de contrôle d'encombrement sont nécessaires (EXIGÉS pour le fonctionnement général de l'Internet) le taux de transmission de l'envoyeur DEVRA être contrôlé par le mécanisme de contrôle d'encombrement. Dans tous les cas, il est RECOMMANDÉ que toutes les transmissions de données provenant d'envoyeurs de diffusion groupée soient soumises aux limitations de débit déterminées par l'algorithme d'application ou de contrôle d'encombrement. Les transmissions de l'envoyeur DEVRAIENT faire une bonne utilisation de la capacité disponible (qui peut être limitée par l'application et/ou le contrôle d'encombrement). Par suite, il est supposé qu'il va y avoir un chevauchement et un multiplexage des nouvelles transmissions de contenu de données avec le contenu de réparation. D'autres facteurs relatifs au fonctionnement de l'application peuvent déterminer les formats et méthodes de transmission de l'envoyeur. Par exemple, une certaine considération doit être apportée au comportement de l'envoyeur durant les périodes d'inactivité intermittentes quand il n'a pas de données à transmettre.

En plus du contenu de données, d'autres messages ou commandes d'envoyeur peuvent être employés au titre du fonctionnement du protocole. Ces messages peuvent survenir en dehors du domaine du transfert de données d'application. Dans les protocoles de diffusion groupée fiable fondée sur le NACK, la fiabilité de tels messages de protocole peut être tentée par une transmission redondante quand l'accusé de réception positif est d'un coût prohibitif à cause des soucis d'adaptabilité de la taille de groupe. Noter que la conception du protocole DEVRAIT fournir des mécanismes pour traiter les cas où de tels messages ne sont pas reçus par le groupe. Par exemple, un message de commande pourrait être transmis de façon redondante par un envoyeur pour indiquer qu'il arrête temporairement (ou de façon permanente) la transmission. À ce moment, il peut être approprié pour les receveurs de répondre avec des NACK pour toute réparation en instance qu'ils exigent, suivant les règles de la procédure de NACK. Pour l'efficacité, l'envoyeur devrait permettre un temps suffisant entre les transmissions redondantes pour recevoir toutes les réponses de NACK provenant des receveurs à cette commande.

En général, quand il y a une opération de NACK résultante ou autre rétroaction, le rythme de transmission redondante de messages de contrôle produits par un envoyeur et autres temporisations de protocole de diffusion groupée fiable fondée sur le NACK devrait dépendre de la plus grande estimation de délai d'aller-retour (GRTT, *Greatest Round-Trip Timing*) et de tout NACK résultant attendu ou autre opération de rétroaction. Le GRTT d'envoyeur est une estimation du pire cas de délai d'aller-retour d'un envoyeur donné à tous receveurs du groupe. Il est supposé que l'intervalle de GRTT est une estimation prudente de l'étendue maximale (par rapport au délai) du groupe de diffusion groupée à travers une topologie de réseau par rapport à un envoyeur donné. Les instances de diffusion groupée fiable fondée sur le NACK DEVRAIENT être capables de s'adapter dynamiquement à une large gamme de topologies de réseaux de diffusion groupée.

Entrées :
1. Données et commande d'application.
2. Identifiant de nœud envoyeur.
3. Identifiants de données.
4. Paramètres de segmentation et de FEC.
5. Taux de transmission.
6. Contrôles d'application.
7. Messages de rétroaction du receveur (par exemple, des NACK).

Résultats :
1. Transmission contrôlée des messages avec des en-têtes qui identifient de façon univoque le contenu de données ou de

réparation dans le contexte de la session de diffusion groupée fiable.
2. Commandes indiquant l'état de l'envoyeur ou les autres actions de contrôle de transport à effectuer.

3.2 Processus de réparation de NACK

Un composant critique des protocoles de diffusion groupée fiable fondée sur le NACK est le processus de réparation de NACK. Cela inclut à la fois le rôle du receveur pour détecter et demander les réparations nécessaires et la réponse de l'envoyeur à de telles demandes. Il y a quatre éléments principaux dans le processus de réparation de NACK :
1. initiation du processus de NACK au receveur,
2. suppression de NACK,
3. contenu du message NACK,
4. traitement du NACK par l'envoyeur et réponse de réparation.

3.2.1 Initiation du processus de NACK au receveur

Le processus (cycle) de NACK va être initié par les receveurs qui détectent un besoin de transmissions de réparation de la part d'un envoyeur spécifique pour réaliser une réception fiable. Quand la FEC est appliquée, un receveur devrait initier le processus de NACK seulement quand il est connu que ses exigences de réparation excèdent la quantité de transmission de FEC en cours pour un bloc de codage donné de contenu de données. Cela peut être déterminé à la fin du bloc de transmission en cours (si elle est indiquée) ou au début de la réception d'un bloc de codage ou objet de transmission suivant. Cela implique que le contenu de données de l'envoyeur est marqué pour identifier son numéro de bloc de FEC et que la relation d'ordre est préservée dans l'ordre de transmission.

Autrement, si la transmission de l'envoyeur annonce la quantité de paquets de réparation qu'il a déjà programmée d'envoyer pour un bloc, le receveur peut être capable d'initier plus tôt le processus de NACK. Permettre aux receveurs d'initier des cycles de NACK à tout moment où ils détectent que leurs besoins de réparation ont excédé les transmissions de réparation en cours peut résulter en des cycles de réparation légèrement plus rapides. Cependant, il peut être utile de limiter l'initiation de processus de NACK à des événements spécifiques, comme la fin de transmission d'un bloc de codage de FEC ou à la détection de blocs de codage suivants. Cela peut permettre aux receveurs d'agréger le contenu de NACK dans un plus petit nombre de messages de NACK et de fournir une synchronisation lâche implicite parmi l'ensemble de receveurs pour aider à faciliter l'efficacité de la suppression probabiliste de rétroactions de NACK. Le receveur DOIT conserver un historique du contenu de données reçues de l'envoyeur pour déterminer ses besoins courants de réparation. Quand la FEC est employée, il est supposé que l'historique va correspondre à un enregistrement des blocs de codage en cours ou partiellement reçus.

Pour la suppression probabiliste fondée sur la temporisation des rétroactions, le cycle de NACK devrait commencer par l'observation par les receveurs de retards de temporisation. En conjonction avec l'initiation de ces retards de temporisation, il est important que les receveurs enregistrent la position dans la séquence de transmission de l'envoyeur à laquelle ils initient le cycle de NACK. Quand arrive à expiration la temporisation de retard de suppression, les receveurs devraient seulement considérer leurs besoins de réparation jusqu'à cette position de transmission enregistrée pour prendre la décision de transmettre ou supprimer un NACK. Sans cette restriction, la suppression est largement réduite lorsque du contenu supplémentaire est reçu de l'envoyeur pendant le temps qu'un message de NACK se propage à travers le réseau à l'envoyeur et aux autres receveurs.

Entrées :
1. Contenu de données d'envoyeur avec séquençage des identifiants provenant des transmissions de l'envoyeur.
2. Historique des contenus reçus de l'envoyeur.

Résultats :
1. Décision d'initiation du processus de NACK.
2. Position enregistrée de la séquence de transmissions de l'envoyeur.

3.2.2 Suppression de NACK

Un mécanisme efficace de suppression de rétroactions est l'utilisation de temporisations aléatoires de retard avant la transmission de NACK par les receveurs qui exigent des réparations [SrmFramework]. À l'expiration de la temporisation de retard, un receveur va demander des réparations sauf si sa réparation en cours a besoin d'être complètement outrepassée par des messages de NACK entendus d'autres receveurs (quand les receveurs envoient les NACK en diffusion groupée) ou à partir d'une indication de l'envoyeur. Quand les receveurs envoient les messages de NACK en envoi individuel,

l'envoyeur peut faciliter la suppression de NACK en transmettant une représentation du contenu de NACK qu'il a reçu du groupe au sens large ou en fournissant un autre indicateur des informations de réparation qu'il va transmettre ensuite.

Pour des performances de suppression effective et adaptable, les périodes de temporisation de retard utilisées par les receveurs devraient être indépendantes, prises au hasard par les receveurs avec une distribution exponentielle tronquée [McastFeedback]. Il en résulte que la majorité de l'ensemble de receveurs retient la transmission de messages de NACK sous l'hypothèse que le plus petit nombre d'envoyeur de "NACK précoces" va surpasser les besoins de réparation du reste du groupe. Le moyen de distribution devrait être déterminé par une fonction de l'affirmation de l'estimation actuelle du GRTT de l'envoyeur et d'une estimation de la taille de groupe qui est soit déterminée par d'autres mécanismes du protocole ou soit préétablie par l'application de diffusion groupée.

Un simple algorithme peut être construit pour générer des temporisations de retard aléatoire avec la distribution appropriée. De plus, l'algorithme peut être conçu pour optimiser la distribution du retard en fonction du nombre de receveurs ("R") générant potentiellement des rétroactions. Cette "optimisation" minimise le nombre de messages de rétroaction (par exemple, de NACK) dans la pire des situations où tous les receveurs génèrent un NACK. La temporisation maximale de retard ("T_maxBackoff") peut être réglée à contrôler la latence de livraison fiable contre le volume de trafic de rétroaction. Une plus grande valeur de "T_maxBackoff" va résulter en une plus faible densité de trafic de rétroaction pour un cycle de réparation donné. Une plus petite valeur de "T_maxBackoff" résulte en une plus courte latence, qui réduit aussi les exigences de mise en mémoire tampon des envoyeurs et receveurs pour un transport fiable.

Dans les fonctions ci-dessous, la fonction "log()" spécifiée se réfère au "logarithme naturel" et la fonction "exp()" est de même fondée sur la constante mathématique "e" (autrement dit le nombre de Euler) où "exp(x)" correspond à "e" à la puissance "x". Étant donné que l'estimation de la taille de groupe du receveur ("tailleDeGroupe") et la temporisation maximum de retard permise ("T_maxBackoff"), des temporisations de retard aléatoires ("t'") avec une distribution exponentielle tronquée peut être prise avec l'algorithme suivant :

1. Établir une moyenne optimale ("L") pour le retard exponentiel fondé sur la "tailleDeGroupe" :

   L = log(tailleDeGroupe) + 1

2. Prendre un nombre aléatoire ("x") dans une distribution uniforme sur une gamme de :

$$\frac{L}{T\_maxBackoff*(exp(L)-1)} \text{ à } \frac{L}{T\_maxBackoff*(exp(L)-1)} + \frac{L}{T\_maxBackoff}$$

3. Transformer la variable aléatoire pour générer le temps de retard aléatoire désiré ("t'") avec l'équation suivante :

   t' = T_maxBackoff/L * log(x * (exp(L) - 1) * (T_maxBackoff/L))

Cette fonction de langage "C" peut être utilisée pour générer un intervalle approprié de retard aléatoire :

```
double RandomBackoff(double T_maxBackoff, double tailleDeGroupe)
{
   double lambda = log(tailleDeGroupe) + 1;
   double x = UniformRand(lambda/T_maxBackoff) + lambda / (T_maxBackoff*(exp(lambda)-1));
      retourne ((T_maxBackoff/lambda) * log(x*(exp(lambda)-1)*(T_maxBackoff/lambda)));
}                                                    // fin de RandomBackoff()
```

où "UniformRand(double max)" retourne des nombres aléatoires avec une distribution uniforme dans la gamme de "0..max". Par exemple, sur la base de la fonction POSIX "rand()", le code "C" suivant peut être utilisé :

```
double UniformRand(double max)
{
retourne (max * ((double)rand()/(double)RAND_MAX));
}
```

Le nombre attendu de messages de NACK générés ("N") dans le premier délai d'aller-retour pour un seul événement de rétroaction est approximativement :

N = exp(1.2 * L / (2*T_maxBackoff/GRTT))

Donc, le temps maximum de retard peut être ajusté pour faire un compromis entre le pire cas de volume de NACK de rétroaction et la latence. Ceci est dérivé des équations données dans [McastFeedback] et suppose que "T_maxBackoff ≥ GRTT", et "L" est la moyenne de la distribution optimisée pour l'estimation donnée de la taille de groupe comme montré dans l'algorithme ci-dessus. Noter que d'autres mécanismes dans le protocole peut travailler à réduire encore la génération redondante de NACK. Il est suggéré que "T_maxBackoff" soit choisi comme un entier multiple de l'estimation courante de GRTT annoncée par l'envoyeur telle que :

T_maxBackoff = K * GRTT ; où K ≥ 1

Pour le fonctionnement général de l'Internet, une valeur par défaut de "K=4" est RECOMMANDÉE pour les opérations de livraison de NACK en diffusion groupée (au groupe au sens large) ; une valeur de "K=6" est RECOMMANDÉE par défaut pour la livraison de NACK en envoi individuel. D'autres valeurs peuvent être utilisées pour réaliser l'utilisation du compromis désiré de mémoire tampon, de latence de livraison fiable, et d'adaptabilité de la taille de groupe.

Étant donné que ("K*GRTT") est le temps maximum de retard utilisé par les receveurs pour initier la transmission de NACK, d'autres périodes de temporisation relatives au processus de réparation de NACK peuvent être adaptées en conséquence. Une de ces temporisations est la durée pendant laquelle un receveur devrait attendre après avoir généré un message de NACK avant de se permettre d'initier un autre cycle de retard/transmission de NACK ("T_rcvrHoldoff"). Ce délai devrait être suffisant pour que l'envoyeur réponde au NACK reçu avec des messages de réparation. Une valeur appropriée dépend du temps nécessaire au NACK pour atteindre l'envoyeur et à l'envoyeur pour fournir une réponse de réparation. Cela DOIT inclure toute période d'agrégation de NACK d'envoyeur durant laquelle plusieurs NACK possibles sont accumulés pour déterminer une réponse de réparation efficace. Ces temporisations sont discutées au paragraphe 3.2.4.

Il y a aussi des mesures secondaires qui peuvent être appliquées pour améliorer les performances de la suppression de rétroactions. Par exemple, les transmissions de contenu de données de l'envoyeur peuvent suivre une séquence ordonnée de transmissions. Quand des réparations de contenu de données se produisent, le receveur peut noter que l'envoyeur a "réorganisé" sa position de transmission de contenu de données en observant l'objet de données, le numéro de bloc de FEC, et les identifiants de symbole de FEC. Les receveurs DEVRAIENT limiter la transmission des NACK à seulement quand la position de transmission courante de l'envoyeur excède le point auquel le receveur a une réception incomplète. Cela réduit les demandes prématurées de réparation de données que l'envoyeur peut avoir prévu de fournir en réponse aux autres demandes de receveurs. Ce mécanisme peut être très efficace pour la convergence de protocole dans des conditions de pertes importantes quand des transmissions de NACK provenant d'autres receveurs (ou indicateurs provenant de l'envoyeur) sont perdues. Un autre mécanisme (particulièrement applicable quand la FEC est utilisée) est que l'envoyeur incorpore une indication de transmissions de réparation imminentes dans les paquets actuellement envoyés. Par exemple, l'indication peut être un simple avertissement du nombre de paquets de FEC à envoyer dans le bloc de codage actuellement applicable.

Finalement, une certaine considération pourrait être accordée à l'utilisation de l'historique des NACK des receveurs pour biaiser leur choix des intervalles de temporisation de retard de NACK. Par exemple, si un receveur a historiquement subi le plus fort degré de pertes, il peut se promouvoir statistiquement plus tôt en NACK que les autres receveurs. Noter que cela exige une corrélation sur les intervalles de temps successifs dans les pertes subies par un receveur. Une telle corrélation PEUT ne pas toujours être présente dans les réseaux de diffusion groupée. Cet ajustement du choix de temporisation de retard peut exiger la création d'un créneau de "NACK précoce" pour ces historiques de NACK. Ce créneau supplémentaire dans la fenêtre de retard de NACK va résulter en un plus long processus de cycle de réparation qui peut n'être pas désirable pour certaines applications. La résolution de ces compromis peut dépendre de l'ensemble d'applications cibles du protocole ou du réseau.

Après l'expiration de la temporisation aléatoire de retard, le receveur va prendre une décision sur si il génère une demande de réparation de NACK ou non (c'est-à-dire, si il a été supprimé). Le NACK va être supprimé quand une des conditions suivante se produit :

1. L'état accumulé des NACK entendus des autres receveurs (ou la transmission de cet état par l'envoyeur) est égal ou dépasse les besoins de réparation du receveur local. Noter que le receveur local devrait considérer ses besoins de réparation seulement jusqu'à la position de transmission de l'envoyeur enregistrée à l'initialisation du cycle de NACK (quand le temporisateur de retard a été activé).

2. La position de transmission du contenu de données de l'envoyeur "revient" à un point ordinal inférieur à celui de la plus basse position de séquence des besoins de réparation du receveur local. (Cette détection d'un "retour" de l'envoyeur indique que l'envoyeur a déjà répondu aux besoins de réparation d'un autre receveur dont le receveur local peut n'avoir

pas été averti). Cet événement de "retour" peut se produire à tout moment entre 1) quand le cycle de NACK a été initialisé avec l'activation de la temporisation de retard et 2) le moment actuel quand la temporisation de retard a expiré pour supprimer le NACK. Un autre cycle de NACK doit être initié par le receveur quand la position de séquence de transmission de l'envoyeur excède le plus bas point ordinal de réparation du receveur. Noter qu'il est possible que le receveur local puisse avoir eu ses besoins de réparation satisfaits par suite de la réponse de l'envoyeur aux besoins de réparation d'autres receveurs et qu'aucun autre NACK ne soit exigé.

Si ces conditions ne se sont pas produites et si le receveur a encore de besoins de réparation en instance, un message NACK est généré et transmis. Le NACK devrait consister en une accumulation de besoins de réparation depuis le plus bas point ordinal de réparation du receveur jusqu'à la position actuelle de la séquence de transmission de l'envoyeur. Un seul message NACK devrait être généré et le contenu du message NACK devrait être tronqué si il excède la taille de charge utile d'un seul message de protocole. Quand une telle limite de charge utile de NACK se produit, le contenu du NACK DEVRAIT contenir des demandes pour le contenu de réparation ordinalement le plus bas nécessaire de l'envoyeur.

Entrées :
1. Décision d'initiation du processus de NACK.
2. Position enregistrée de la séquence de transmission de l'envoyeur.
3. GRTT de l'envoyeur.
4. Estimation de la taille de groupe de l'envoyeur.
5. Limite définie par l'application sur la période de temporisation de retard.
6. Les NACK provenant des autres receveurs.
7. Indication de réparation en instance de l'envoyeur (peut être les NACK transmis).
8. Position actuelle de la séquence de transmission de l'envoyeur.

Résultats :
1. Décision oui/non de générer un message NACK à l'expiration du temporisateur de retard.

3.2.3 Contenu du message NACK

Le contenu des messages de NACK générés par des receveurs de diffusion groupée fiable va inclure des informations détaillant leurs besoins actuels de réparation. Les informations spécifiques dépendent de l'utilisation et du type de FEC dans le processus de réparation de NACK. L'identification des besoins de réparation dépend de l'identification du contenu de données (voir le paragraphe 3.5). Au plus haut niveau, le contenu de NACK va identifier l'envoyeur auquel le NACK est adressé et l'objet (ou flux) de transport de données dans la transmission de l'envoyeur qui a besoin de réparation. Pour l'entité de transport indiquée, le contenu de NACK va alors identifier les blocs de codage de FEC spécifiques et/ou les symboles qu'il exige pour reconstruire les données transmises complètes. Ce contenu peut consister en comptes d'écrasement de blocs de FEC et/ou en l'indication explicite des blocs ou symboles (segments) manquants de données et de contenu de FEC. On devrait aussi noter que la diffusion groupée fiable fondée sur le NACK peut être effectivement instanciée sans l'exigence d'une livraison fiable de NACK en utilisant les techniques discutées ici.

3.2.3.1 NACK et stratégies de réparation de FEC

Lorsque la réparation fondée sur la FEC est utilisée, le contenu du message de NACK va au minimum devoir identifier le ou les blocs de codage pour lesquels la réparation est nécessaire et un compte des suppressions (paquets manquants) pour le bloc de codage. Un compte exact des suppressions implique que l'algorithme de FEC soit capable de réparer toutes les combinaisons de pertes au sein du bloc de codage. Ce compte peut devoir être ajusté pour certains algorithmes de FEC.

Considérant que plusieurs tours de réparation peuvent être exigés pour achever avec succès la réparation, un compte de suppressions implique aussi que la quantité de paquets de parité unique de FEC que le serveur a disponibles à transmettre est essentiellement illimitée (c'est-à-dire, le serveur va toujours être capable de fournir de nouveaux paquets, uniques, précédemment non envoyés, en réponse à toute demande de réparation suivante pour le même bloc de codage). Autrement, l'envoyeur peut faire une transmission "round-robin" à travers l'ensemble disponible de symboles de FEC pour un bloc de codage donné, et finalement effectuer la réparation. Pour la stratégie de réparation la plus efficace, le contenu de NACK va devoir aussi identifier explicitement quels symboles (information et/ou parité) le receveur exige pour réussir à reconstruire le contenu du bloc de codage. Cela va être particulièrement vrai des codes de FEC de bloc de petite et moyenne taille (par exemple, Reed Solomon [RFC5510]) qui sont capables de fournir un nombre limité de symboles de parité par bloc de codage de FEC.

Quand la FEC n'est pas utilisée au titre du processus de réparation, ou quand il est exigé de l'instance de protocole qu'elle assure la fiabilité même quand l'envoyeur a transmis toute la parité disponible pour un certain bloc de codage (ou quand la

capacité de l'envoyeur de mettre en mémoire tampon l'historique de transmission est excédée par les caractéristiques de "(délai*bande-passante*perte)" de la topologie du réseau) le contenu de NACK va devoir contenir des informations explicites de bloc de codage et/ou de perte de segment afin que l'envoyeur puisse fournir des paquets de réparation appropriés et/ou des retransmissions de données. Des informations explicites de perte dans le contenu de NACK peuvent aussi éventuellement servir à d'autres fins. Par exemple, elles peuvent être utiles pour décorréler les caractéristiques de perte parmi un groupe de receveurs pour aider à différencier les goulets d'étranglement candidats de contrôle d'encombrement parmi l'ensemble de receveurs.

Quand la FEC est utilisée et que le contenu de NACK est destiné à contenir des demandes explicites de réparation, il y a une stratégie où les receveurs peuvent envoyer un NACK pour un contenu spécifique qui va aider à faciliter la suppression de NACK et l'efficacité de la réparation. Les hypothèses de cette stratégie sont que l'envoyeur peut épuiser potentiellement sa fourniture de nouveaux paquets uniques de parité disponibles pour un certain bloc de codage et qu'il soit exigé qu'il retransmette explicitement des données ou symboles de parité pour réaliser un transfert fiable. Une autre hypothèse est qu'est utilisé un algorithme de FEC où un paquet de parité peut combler une suppression au sein du bloc de codage (par exemple, Reed Solomon). Le but de cette stratégie est de faire une utilisation maximale de la parité disponible et de fournir la quantité minimale de données et de transmissions de réparation durant le transfert fiable du contenu de données au groupe.

Quand les codes de FEC systématiques sont utilisés, l'envoyeur transmet le contenu de données du bloc de codage (et facultativement une certaine quantité de paquets de parité) dans sa transmission initiale. Noter qu'un bloc de codage de FEC systématique est considéré être logiquement constitué d'un ensemble contigu de vecteurs de données de source plus des vecteurs de parité pour l'algorithme de FEC utilisé. Par exemple, un schéma de codage systématique qui fournit 64 symboles de données et 32 symboles de parité par bloc de codage contiendrait des identifiants de symboles de FEC dans la gamme de 0 à 95.

Les receveurs peuvent alors construire des messages de NACK demandant un contenu suffisant pour satisfaire leurs besoins de réparation. Par exemple, si le receveur a trois suppressions dans un certain bloc de codage reçu, il va demander la transmission des trois vecteurs de parité de plus bas ordre dans le bloc de codage. Dans notre exemple de schéma de codage du paragraphe précédent, le receveur demanderait explicitement les symboles de parité 64 à 66 pour remplir ses trois suppressions pour le bloc de codage. Noter que si les pertes du receveur pour le bloc de codage excèdent la quantité de parité disponible (c'est-à-dire, supérieures à 32 symboles manquants dans notre exemple) le receveur va devoit construire un NACK demandant tous (32) les symboles de parité disponibles plus des portions supplémentaires de ses symboles de données manquants afin de reconstruire le bloc. Si cela est fait de façon cohérente à travers le groupe de receveurs, les NACK résultants vont comprendre un ensemble minimal de transmissions de l'envoyeur pour satisfaire les besoins de réparation.

En résumé, la règle est de demander la portion ordinale la plus basse du contenu de parité pour le bloc de codage de FEC pour satisfaire les besoins de réparation de suppression sur le premier cycle de NACK. Si le nombre disponible de symboles de parité est insuffisant, le receveur va aussi demander le sous ensemble de symboles de données manquantes d'ordre supérieur pour couvrir ce que les symboles de parité ne vont pas remplir. Noter que cette stratégie suppose des codes de FEC comme Reed-Solomon pour lesquels un seul symbole de parité peut réparer tout symbole supprimé. Cette stratégie nécessiterait une modification mineure pour tenir compte de la possibilité d'une capacité de réparation limitée des autres types de FEC. Sur les cycles suivants de réparation de NACK où le receveur peut recevoir une portion de son contenu de réparation précédemment demandé, le receveur va utiliser la même stratégie, mais va seulement faire des NACK pour l'ensemble de symboles de parité et/ou de données qu'il n'a pas encore reçus. Facultativement, les receveurs pourraient aussi fournir un compte des suppressions comme une faveur à l'envoyeur.

D'autres types de schémas de FEC peuvent exiger l'altération de la stratégie de NACK et de réparation décrite ici. Par exemple, certains des codes de FEC de grand bloc ou extensibles décrits dans la [RFC3453] peuvent être moins déterministes à l'égard de la définition des demandes de réparation optimale par les receveurs ou des stratégies de transmission de réparation par les envoyeurs. Pour ces types de codes, il peut être suffisant que les receveurs envoient un NACK avec une estimation de la quantité de symboles de FEC supplémentaires exigée pour réaliser la réception fiable et pour que l'envoyeur réponde en conséquence. Ce désavantage apparent, comparé aux codes comme Reed Solomon, peut être effacé par la réduction des exigences de calcul et/ou la capacité de prendre en charge de grands blocs de codage pour une efficacité accrue de réparation que ces codes peuvent offrir.

Après la réception et l'accumulation de messages de NACK durant la période d'agrégation, l'envoyeur peut commencer la transmission de symboles de parité frais (non transmis précédemment) pour le bloc de codage sur la base du plus fort compte de suppression du receveur si il a une quantité suffisante de symboles de parité qui n'ont pas été transmis précédemment. Autrement, l'envoyeur DOIT se résoudre à transmettre l'ensemble explicite de vecteurs de réparation demandé. Avec cette approche, l'envoyeur a besoin de conserver très peu d'état sur les demandes qu'il a reçues du groupe

sans avoir besoin de synchronisation des demandes de réparation provenant du groupe. Comme tous les receveurs utilisent le même algorithme cohérent pour exprimer leurs besoins explicites de réparation, la suppression de NACK parmi les receveurs est simplifiée sur le cours de plusieurs cycles de réparation. Les receveurs peuvent simplement comparer les NACK entendus des autres receveurs à leurs propres besoins de réparation calculés pour déterminer si ils devraient transmettre ou supprimer leurs messages de NACK en instance.

3.2.3.2 Format du contenu de NACK

Le format du contenu de NACK va dépendre du modèle de service de données du protocole et du format d'identification de contenu de données qu'utilise le protocole. Ce format de NACK dépend aussi du type de codage de FEC (si il y en a un) utilisé. La Figure 2 illustre un schéma logique d'identification de contenu de transmission hiérarchique, qui note que la notion d'objets (ou de flux) et/ou de mise en blocs de FEC est facultatif à la discrétion de l'instance de protocole. Noter que l'identification des objets est par rapport à un certain envoyeur. Il est recommandé que l'identification du contenu de données de transport soit faite dans le contexte d'un envoyeur dans une session donnée. Comme la notion de "flux" et "blocs" de session est facultative, le cadre revient à celui de la segmentation et ré-assemblage normal de données de transport dans sa plus simple forme.

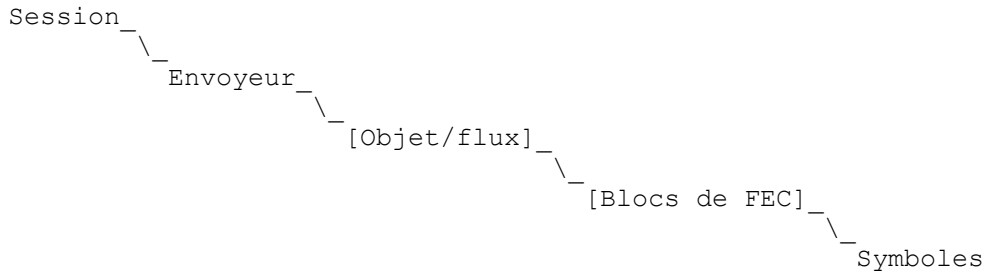


Figure 2 : Hiérarchie d'identification de contenu de données de diffusion groupée fiable

Le format de messages de NACK devraient permettre ce qui suit :
1. l'identification des unités de données de transport exigées pour la réparation du contenu reçu, que ce soit un objet/flux manquant entier (ou gamme) un ou des blocs de codage de FEC entiers, ou des ensembles de symboles,
2. le simple traitement d'agrégation et suppression de NACK,
3. l'inclusion de NACK pour plusieurs objets, blocs de codage de FEC, et/ou symboles dans un seul message, et
4. un format raisonnablement compact.

Si l'objet/flux de transport de diffusion groupée fiable est identifié avec un <objectId> et si le symbole de FEC transmis est identifié avec un <fecPayloadId>, l'enchaînement de <objectId::fecPayloadId> comprend un identifiant de base d'unité de données de protocole de transport (TPDU, *Transport Protocol Data Unit*) pour les symboles provenant d'une certaine source. Le contenu de NACK peut être composé de listes et/ou de gammes de ces identifiants de TPDU pour construire des messages de NACK décrivant les besoins de réparation du receveur. Si aucune délimitation hiérarchique d'objet ou mise en blocs de FEC n'est utilisée, la TPDU est une simple représentation linéaire des symboles de données transmis par l'envoyeur. Quand la TPDU représente une hiérarchie pour les besoins de la délimitation d'objet/flux et/ou la mise en blocs de FEC, l'unité de contenu de NACK peut exiger des fanions pour indiquer quelle portion de la TPDU est applicable. Par exemple, si un "objet" (ou gamme d'objets) entier manque dans les données reçues, le receveur ne va pas nécessairement savoir la gamme appropriée de <sourceBlockNumbers> ou <encodingSymbolIds> pour laquelle demander réparation et donc il faut un mécanisme pour demander la réparation (ou la retransmission) de l'unité entière représentée par un <objectId>. La même chose est vraie si des blocs entiers de codage de FEC représentés par un ou une gamme de <sourceBlockNumbers> ont été perdus.

Entrées :
1. identification de l'envoyeur.
2. identification des données de l'envoyeur.
3. informations de transmission d'objet de FEC de l'envoyeur.
4 position enregistrée de la séquence de transmission de l'envoyeur.
5. position courante de la séquence de transmission de l'envoyeur. Historique des besoins de réparation pour cet envoyeur.

Résultats :
1. message NACK avec demandes de réparation.

3.2.4 Traitement de NACK à l'envoyeur et réponse de réparation

À réception d'une demande de réparation d'un receveur du groupe, l'envoyeur va initier une procédure de réponse de réparation. L'envoyeur peut souhaiter retarder la transmission du contenu de réparation jusqu'à ce qu'il ait eu un temps suffisant pour accumuler éventuellement plusieurs NACK de l'ensemble de receveurs. Cela permet à l'envoyeur de déterminer la stratégie de réparation la plus efficace pour un flux/objet de transport ou bloc de codage de FEC donné. Selon l'approche utilisée, certains protocoles peuvent trouver bénéfique pour l'envoyeur de fournir un indicateur des transmissions de réparation en instance au titre de son contenu de messages actuellement transmis. Cela peut aider certains mécanismes de suppression de NACK. Le temps pour effectuer cette agrégation de NACK devrait être suffisant pour permettre la fenêtre maximum de retard de NACK du receveur (""T_maxBackoff"" du paragraphe 3.2.2) et la propagation des messages de NACK des receveurs à l'envoyeur. Noter que le délai maximum de transmission d'un message d'un receveur à l'envoyeur peut être approximativement de "(1*GRTT)" dans le cas d'une topologie de réseau très asymétrique par rapport au délai de transmission. Donc, si le temps maximum de retard de NACK du receveur est "T_maxBackoff = K*GRTT", la période d'agrégation de NACK de l'envoyeur devraient être égale au moins à :

T_sndrAggregate = T_maxBackoff + 1*GRTT = (K+1)*GRTT

Immédiatement après la période d'agrégation de NACK de l'envoyeur, celui-ci va commencer à transmettre le contenu de réparation déterminé à partir de l'état agrégé de NACK et continuer avec toute nouvelle transmission. Aussi, à ce moment, l'envoyeur devrait observer une période de "pause" pendant laquelle il s'abstient d'initier une nouvelle période d'agrégation de NACK pour permettre la propagation de la nouvelle position de séquence de transmission due à la réponse de réparation au groupe receveur. Pour permettre le pire cas d'asymétrie, ce temps de "pause" devrait être :

T_sndrHoldoff = 1*GRTT

On se rappelle que les receveurs vont aussi employer une temporisation de "pause" après avoir généré un message NACK pour donner du temps à la réponse de l'envoyeur. En donnant un temps de "(K+1)*GRTT" au "<T_sndrAggregate>" plus "<T_sndrHoldoff>" d'un envoyeur, les receveurs devraient utiliser des temporisations de pause de :

T_rcvrHoldoff = T_sndrAggregate + T_sndrHoldoff = (K+2)*GRTT

Cela permet un pire cas de temps de propagation du NACK de receveur à l'envoyeur, le temps d'agrégation de l'envoyeur, et la propagation de la réponse de l'envoyeur en retour au receveur. De plus, dans le cas de rétroaction en envoi individuel de la part de l'ensemble receveur, il peut être utile que l'envoyeur transmette (par diffusion groupée) une représentation de son contenu agrégé de NACK au groupe pour permettre la suppression de NACK quand il n'y a pas de connexité de diffusion groupée parmi l'ensemble de receveurs.

À l'expiration de la temporisation "<T_sndrAggregate>", l'envoyeur va commencer à transmettre ses messages de réparation en accord avec le contenu accumulé de NACK reçus. On donne des lignes directrices sur la réparation fondée sur la FEC et l'ordre de réponse de réparation de l'envoyeur qui peuvent améliorer l'efficacité de diffusion groupée fiable :

Quand la FEC est utilisée, il est bénéfique que l'envoyeur transmette le contenu de parité non transmis précédemment comme messages de réparation chaque fois que possible. Cela maximise la capacité des nœuds receveurs de reconstruire le contenu transmis entier à partir de leurs sous ensembles individuels de messages reçus.

Les données d'objet et/ou flux transmises et le contenu de réparation devraient être indexés avec des numéros de séquence à croissance monotone (dans un espace ordinal raisonnablement grand). Si l'envoyeur observe la discipline de transmission de réparation en premier du contenu le plus ancien (par exemple, les blocs de FEC de numéros les plus faibles) les receveurs peuvent utiliser une stratégie de rétention des demandes de réparation pour un contenu ultérieur jusqu'à ce que l'envoyeur retourne une fois encore à ce point dans la séquence de transmission d'objet/flux. Cela peut augmenter l'efficacité globale de message parmi le groupe et aider à garder les cycles de réparation relativement synchronisés sans dépendre d'une synchronisation stricte entre l'envoyeur et les receveurs. Cela aussi aide à minimiser les exigences de mémoire tampon des receveurs et envoyeurs et réduit les transmissions redondantes de données au groupe au sens large.

Entrées :
1. messages de NACK du receveur.
2. informations de rythme du groupe.

Résultats :
1. messages de réparation (FEC et/ou retransmission de contenu de données).
2. annonce des transmissions de réparation en cours quand une rétroaction en envoi individuel du receveur est détectée.

3.3 Politiques et procédures de jonction de receveur de diffusion groupée

Les politiques et procédures par lesquelles de nouveaux receveurs se joignent à un groupe (peut-être lorsque la transmission fiable est déjà en cours) et commencent à demander des réparations devraient être examinées. Si les jonctions de receveurs sont sans contraintes, la dynamique de l'adhésion au groupe peut entraver la capacité de l'application à satisfaire ses buts de progrès de transmission des données. Les politiques qui limitent les opportunités des receveurs de commencer à participer au processus de NACK peuvent être utilisées pour réaliser le comportement désiré. Par exemple, il peut être bénéfique aux receveurs de ne tenter la réception fiable à partir d'un envoyeur nouvellement entendu que sur des transmissions de données de non réparation dans le premier bloc de FEC d'un objet ou de la portion logique d'un flux. L'envoyeur peut aussi mettre en œuvre des politiques limitant les receveurs de qui il va accepter des demandes de NACK, mais cela peut être prohibitif pour des raisons d'adaptabilité dans certaines situations. Autrement, il peut être souhaitable d'avoir une politique de synchronisation de transport plus souple et de s'appuyer seulement sur des mécanismes de gestion de session pour limiter les dynamiques de groupe qui peuvent causer de mauvaises performances dans certains types d'applications de transfert en vrac (ou pour des applications de diffusion groupée fiable potentiellement interactives).

Entrées :
1. Contenu actuel de données/réparation d'objet/flux et identifiants de séquence des transmissions d'envoyeur.

Résultats :
1. Décision oui/non du receveur de commencer à recevoir et envoyer des NACK pour la réception fiable de données.

3.4 Identification de nœud (membre)

Dans un protocole de diffusion groupée fiable fondée sur le NACK (ou autres protocoles de diffusion groupée) lorsque il y a un potentiel de plusieurs sources de données, il est nécessaire de fournir un mécanisme pour identifier de façon univoque les sources (et éventuellement certains ou tous les receveurs) au sein du groupe. Les receveurs qui envoient des messages de NACK au groupe vont devoir identifier l'envoyeur auquel le NACK est destiné. L'identité fondée sur les adresses de source des paquets arrivants est insuffisante pour plusieurs raisons. Ces raisons incluent des changements d'acheminement pour les hôtes avec plusieurs interfaces qui résultent en différentes adresses de source de paquet pour un hôte donné à un moment donné, d'appareils de traduction d'adresse réseau (NAT, *Network Address Translation*) ou de pare-feu, ou autres approches de pontage de transport/réseau. Par suite, un type de champ d'identifiant unique de source <sourceId> DEVRAIT être présent dans les paquets transmis par les membres des sessions de diffusion groupée fiable.

3.5 Identification du contenu de données

Le contenu de données et de réparation transmis par un envoyeur de diffusion groupée fiable fondée sur le NACK exige une forme d'identification dans les champs d'en-tête de protocole. Cette identification est exigée pour faciliter le processus de réparation fiable en mode NACK. Ces identifiants vont aussi être utilisés dans les messages de NACK générés. Le présent document de bloc de construction suppose deux types très généraux de données qui peuvent comprendre du contenu de session de transfert en vrac. Un type est d'objets statiques discrets de taille finie et l'autre est de flux continus non finis. Une application peut souhaiter un contenu de données en diffusion groupée fiable utilisant l'un ou l'autre ou ces deux paradigmes. Bien qu'il soit possible à certaines applications de généraliser encore ce modèle et fournir des mécanismes pour encapsuler les objets statiques comme du contenu incorporé dans un flux, il y a des avantages dans de nombreuses applications à fournir une prise en charge distincte pour les objets statiques en vrac et les messages avec le contexte d'une session de diffusion groupée fiable. Ces applications peuvent inclure des serveurs de mise en antémémoire de contenu, du transfert de fichiers, ou des outils collaboratifs avec du contenu en vrac. Les applications qui ont des exigences pour ces types d'objets statiques peuvent alors tirer parti des mécanismes de couche de transport (c'est-à-dire, la segmentation/ré-assemblage, la mise en antémémoire, le codage intégré d'erreur directe, etc.) plutôt qu'être obligé de fournir leurs propres mécanismes pour ces fonctions à la couche application.

Comme on l'a noté, certaines applications peuvent aussi désirer transmettre le contenu en vrac sous la forme d'un ou plusieurs flux de taille non finie. Des exemples de flux incluent de diffuser des messages en continu en temps quasi réel (par exemple, des cours de bourse) ou des types de contenu qui font partie d'outils collaboratifs ou d'autres applications. Et comme indiqué ci-dessus, certaines applications peuvent souhaiter encapsuler d'autre contenu en vrac (par exemple, des fichiers) dans un ou plusieurs flux au sein d'une session de diffusion groupée.

Les composants décrits dans le présent document de blocs de construction sont envisagés comme applicables à ces deux modèles avec un mélange potentiel des deux types au sein d'une seule session de diffusion groupée. Pour prendre en charge

cette exigence, l'identification normale du contenu de données devrait inclure un champ qui identifie de façon univoque l'objet ou flux (par exemple, <objectId>) au sein d'un intervalle temporel ou ordinal raisonnable. Noter qu'il n'est pas attendu que cette identification de contenu de données soit unique au monde. On suppose que l'identifiant d'objet/flux va être unique par rapport à un envoyeur au sein de la session de diffusion groupée fiable et durant le temps où l'envoyeur prend en charge une instance de transport spécifique de cet objet ou flux.

Comme un contenu d'objet/flux "en vrac" exige généralement la segmentation, une certaine forme d'identification de segment doit aussi être fournie. Cet identifiant de segment va se rapporter à tout identifiant d'objet ou flux qui a été fourni. Donc, dans certains cas, les instances de protocole de diffusion groupée fiable fondée sur le NACK peuvent être capables de recevoir des demandes de transmissions et de réparations pour plusieurs flux et un ou plusieurs ensembles d'objets statiques en parallèle. Pour les instances de protocole qui emploient la FEC, la portion identification de segment de l'identifiant de contenu de données peut consister en l'enchaînement logique d'un identifiant de bloc de codage <sourceBlockNumber> et d'un identifiant pour les données ou symboles de parité spécifiques <encodingSymbolId> du bloc de code. Le bloc de construction "Schémas de base de FEC" [RFC5445] et les descriptions de schémas de FEC supplémentaires qui pourront être documentés plus tard fournissent un format de message standard pour identifier le contenu de transmission de FEC. Les instances de protocole de diffusion groupée fiable fondée sur le NACK qui utilisent la FEC DEVRAIENT suivre ces lignes directrices.

De plus, des fanions pour déterminer l'usage des champs d'identifiant de contenu (par exemple, de flux ou d'objet) peuvent être applicables. Les fanions peuvent aussi servir à d'autres fins dans l'identification de contenu de données. On s'attend à ce que tous les fanions définis dépendent des instances individuelles de protocole.

En résumé, les champs d'identification de contenu de données suivants peuvent être exigés pour les messages de contenu de données de protocole de diffusion groupée fiable fondée sur le NACK :

1. Identifiant de nœud source (<sourceId>).
2. Identifiant d'objet/flux (<objectId>) si applicable.
3. Identifiant de bloc de FEC (<sourceBlockNumber>) si applicable.
4. Identifiant de symbole de FEC (<encodingSymbolId>).
5. Fanions pour différencier l'interprétation des champs d'identifiant ou la structure d'identifiant qui indiquent implicitement l'usage.
6. Champs supplémentaire de contenu de transmission de FEC par bloc de construction de FEC.

Ces champs ont été identifiés parce que tout message de NACK généré va utiliser ces identifiants pour demander la réparation ou la retransmission de données.

3.6 Correction d'erreur directe (FEC)

Plusieurs approches de la correction d'erreur directe (FEC) utilisant les techniques de codage de suppression ont été identifiées et peuvent fournir de grandes améliorations de performances au processus de réparation en mode NACK et autres protocoles de diffusion groupée fiable [FecBroadcast], [RmFec], [RFC3453]. Les protocoles de diffusion groupée fiable fondée sur le NACK peuvent rapporter des bénéfices supplémentaires dans la mesure où la réparation fondée sur la FEC n'exige généralement pas de connaissance explicite du contenu de réparation dans les limites de sa taille de bloc de codage (en symboles). Dans la diffusion groupée fiable fondée sur le NACK, les paquets de réparation de parité générés vont généralement être transmis seulement en réponse aux demandes de réparation de NACK provenant des nœuds receveurs. Cependant, il y a des avantages dans certains environnements de réseau pour la transmission d'une certaine quantité prédéterminée de paquets de réparation de FEC multiplexés avec les transmissions régulières de symboles de données [FecHybrid]. Cela peut réduire la quantité de trafic de NACK généré avec relativement peu de frais généraux quand les tailles de groupes sont très grandes ou que la connexité au réseau a un gros produit "délai*bande passante" avec un niveau nominal de perte de paquets attendu. Bien que l'application de FEC ne soit pas particulière à la diffusion groupée fiable fondée sur le NACK, ces sortes d'exigences peuvent dicter les types d'algorithmes et d'approches de protocole applicables.

Une question spécifique de l'utilisation de la FEC avec la diffusion groupée fiable fondée sur le NACK est le mécanisme utilisé pour identifier la ou les portions de contenu de données transmises auxquelles les paquets spécifiques de FEC sont applicables. On s'attend à ce que les algorithmes de FEC seront fondés sur la génération d'un ensemble de paquets de réparation de parité pour un bloc correspondant de paquets de données transmis. Comme les paquets de contenu de données sont identifiés de façon unique par l'enchaînement de <sourceId::objectId:: sourceBlockNumber::encodingSymbolId> durant le transport, on s'attend à ce que les paquets de FEC soient identifiés de manière similaire. Le document de blocs de construction de FEC [RFC5052] fournit des recommandations détaillées concernant l'application des formats de FEC et de formats standard pour les messages de protocole de diffusion groupée fiable en relation.

3.7 Collecte de temps d'aller-retour

Il est exigé que la mesure du temps d'aller-retour (RTT) de la propagation de paquet parmi les membres du groupe prenne en charge des algorithmes de suppression de NACK fondés sur le temporisateur, le rythme des commandes de l'envoyeur ou certaines fonctions de réparation, et les opérations de contrôle d'encombrement. La nature des informations d'aller-retour collectées dépend du type d'interaction entre les membres du groupe. Dans le cas d'une transmission de "un à plusieurs", il se peut que seul l'envoyeur exige la connaissance du RTT, du GRTT, et/ou la connaissance du RTT de seulement une portion du groupe. Ici, les informations de GRTT pourraient être collectées d'une manière raisonnablement adaptable. Pour le fonctionnement du contrôle d'encombrement, il est possible que chaque receveur dans le groupe puisse avoir besoin de connaître son RTT individuel. Dans ce cas, un autre schéma de collecte de RTT peut être utilisé lorsque les receveurs collectent les mesures individuelles de RTT par rapport à l'envoyeur et les annoncent au groupe ou à l'envoyeur. Lorsque il est probable que l'échange des données de diffusion groupée fiable va se produire parmi le groupe sur la base de "plusieurs à plusieurs", il y a des techniques de mesure de remplacement qui pourraient être employées pour augmenter l'efficacité [DelayEstimation]. Dans certains cas, il peut y avoir une synchronisation absolue disponible parmi les hôtes participants qui peut simplifier la mesure de RTT. Il y a des compromis dans la conception du contrôle d'encombrement de diffusion groupée qui exigent une certaine attention avant une recommandation universelle sur la façon dont la mesure de RTT (ou GRTT) peut être spécifiée. Sans considération de la façon dont les informations de RTT sont collectées (et plus spécifiquement du GRTT) par rapport au contrôle d'encombrement ou autres exigences, l'envoyeur va devoir annoncer son estimation actuelle de GRTT au groupe pour les diverses temporisations de NACK utilisées par les receveurs.

3.7.1 Mesure de GRTT d'envoyeur de un à plusieurs

Le but de cette forme de mesure de RTT est que l'envoyeur estime le GRTT parmi les receveurs qui participent activement au fonctionnement de la diffusion groupée fiable fondée sur le NACK. L'ensemble de receveurs participants à ce procès peut être le groupe entier ou un sous ensemble du groupe déterminé à partir d'un autre mécanisme au sein de l'instance de protocole. Voici une approche pour collecter ces informations de GRTT.

L'envoyeur interroge périodiquement le groupe avec un message (indépendant ou "porté" avec d'autres transmissions) contenant un horodatage "<sendTime>" relatif à une horloge interne de l'envoyeur. À réception de ce message, les receveurs vont enregistrer cet horodatage "<sendTime>" et l'heure (référencée à leur propre horloge) à laquelle il a été reçu "<recvTime>". Quand le receveur fournit des rétroactions à l'envoyeur (explicitement ou au titre d'autres messages de rétroaction selon la spécification de l'instance de protocole) il va construire une "réponse" utilisant la formule :

grttResponse = sendTime + (currentTime - recvTime)

où le "<sendTime>" est l'horodatage provenant du dernier message de sondage reçu de la source et le ("<currentTime> - <recvTime>") est le differentiel de temps depuis la réception de la demande jusqu'à ce que le receveur génère la réponse.

L'envoyeur traite chaque réponse de receveur en calculant une mesure actuelle de RTT pour le receveur d'où la réponse a été reçue en utilisant la formule suivante :

RTT_rcvr = currentTime - grttResponse

Durant chaque intervalle périodique de sondage de "GRTT", la source garde la mesure crête de délai d'aller-retour ("RTT_peak") provenant de l'ensemble de réponses reçues. Une estimation prudente du "GRTT" est conservée pour maximiser l'efficacité de l'agrégation redondante de suppression de NACK et de réparation. Les mises à jour de l'estimation courante de la source du "GRTT" sont faites en observant les règles suivantes :

1. Si le délai d'aller-retour de réponse d'un receveur ("RTT_rcvr") est supérieur à l'estimation actuelle du "GRTT", le "GRTT" est immédiatement mis à jour à cette nouvelle valeur de crête :

   GRTT = RTT_rcvr

2. À la fin de la période de collecte des réponses (c'est-à-dire, l'intervalle de sondage de GRTT) si la réponse de "crête" enregistrée ("RTT_peak") est moins que l'estimation courante de GRTT, le GRTT est mis à jour à :

   GRTT = MAX(0,9*GRTT, RTT_peak)

3. Si aucune rétroaction n'est reçue, l'estimation du "GRTT" de l'envoyeur reste inchangée.

4. À la fin de la période de collecte de réponses, la valeur de traçage de crête ("RTT_peak") est remise à zéro pour la détection de crête suivante.

La période de collecte de GRTT (c'est-à-dire, la période de transmission de sondes) pourrait être fixée à une valeur de l'ordre de ce qui est attendu pour les membres du groupe et/ou la dynamique de la topologie de réseau. Pour la robustesse, un sondage plus rapide pourrait être utilisé au démarrage du protocole avant de régler un intervalle moins fréquent, de régime de croisière. Facultativement, un algorithme peut être développé pour ajuster la période de collecte de GRTT dynamiquement en réponse à l'estimation courante de GRTT (ou ses variations) et à une estimation de perte de paquets. Les frais généraux des messages de sondage pourraient alors être réduits quand l'estimation de GRTT est stable et ne change pas, mais être ajustée pour la suivre de façon plus dynamique durant des périodes de variation avec des périodes de collecte de GRTT corrélativement plus courtes. La collecte de GRTT PEUT aussi être couplée avec la collecte d'autres informations pour les besoins du contrôle d'encombrement.

En résumé, bien que les temporisations de cycle de réparation de NACK soient fondées sur le GRTT, on devrait noter que le fonctionnement convergent du protocole ne dépend pas d'une estimation très précise du GRTT . Le mécanisme actuel s'est révélé suffisant dans les simulations et dans les environnements où les protocoles de diffusion groupée fiable fondée sur le NACK ont été déployés à ce jour. L'estimation fournie par cet algorithme suit l'enveloppe de crête du GRTT réel (incluant l'effet de système d'exploitation et les délais du réseau) même dans une perte de connexité relativement élevée. L'intervalle de sondage/mise à jour en régime permanent peut éventuellement varier pour s'accommoder des différents niveaux de dynamique de réseau attendus dans différents environnements.

3.7.2 Mesure de RTT de receveur de un à plusieurs

Dans cette approche, les receveurs envoient des messages avec des horodatages à l'envoyeur. Pour contrôler le volume de ces messages générés par le receveur, un mécanisme de suppression similaire à celui décrit pour la suppression de NACK peut être utilisé. L'âge de la mesure de RTT des receveurs devrait être gardé par les receveurs et utilisé comme métrique dans le concours d'opportunités de rétroactions dans le schéma de suppression. Par exemple, un receveur qui n'a pas fait de mesure de RTT ou dont la mesure de RTT a vieilli le plus devrait avoir la préséance sur les autres receveurs. À son tour, l'envoyeur peut avoir une capacité limitée de fournir un "écho" en retour aux horodatages du receveur au groupe, et il pourrait utiliser cette métrique d'âge de RTT pour déterminer quels receveurs ont la préséance. L'envoyeur peut déterminer le "GRTT" comme décrit en 3.7.1 si il fournit des horodatages d'envoyeur au groupe. Autrement, les receveurs qui notent que leur RTT est supérieur au GRTT de l'envoyeur peuvent concourir dans le schéma de suppression/opportunité de rétroaction pour fournir cette information à l'envoyeur et au groupe.

3.7.3 Mesure de RTT de plusieurs à plusieurs

Pour les sessions de diffusion groupée fiable qui impliquent plusieurs envoyeurs, il peut être utile d'avoir des mesures de RTT pratiquées sur une vraie base de "plusieurs à plusieurs" plutôt que chaque envoyeur retrace indépendamment le RTT. Une certaine efficacité de protocole peut être obtenue quand les receveurs peuvent déduire une approximation de leur RTT par rapport à un envoyeur sur la base des informations de RTT qu'ils ont d'un autre envoyeur et du RTT de l'autre envoyeur par rapport au nouvel envoyeur intéressant. Par exemple, pour le receveur "a" et les envoyeurs "b" et "c", il est probable que :

$$RTT(a<->b) \leq RTT(a<->c)) + RTT(b<->c)$$

Plus de raffinement de cette estimation peut être obtenu si les informations de RTT sont disponibles à un nœud concernant son propre RTT par rapport à un petit sous ensemble des autres membres du groupe et si les informations concernant le RTT parmi ces autres membres du groupe sont apprises par le nœud durant le fonctionnement du protocole.

3.7.4 Annonce de GRTT d'envoyeur

Pour faciliter un fonctionnement déterministe du protocole, l'envoyeur devrait annoncer activement son estimation actuelle de "GRTT" à l'ensemble de receveurs. Une connaissance commune robuste du GRTT actuel de l'envoyeur parmi le groupe va permettre au protocole de progresser de la manière la plus efficace. L'estimation du GRTT de l'envoyeur peut être annoncée de façon robuste au groupe en incorporant simplement l'estimation dans tous les messages pertinents transmis par l'envoyeur. Les frais généraux de cela peuvent être rendus assez petits en quantifiant (compressant) l'estimation de GRTT dans un seul octet d'information. Les fonctions en langage C suivantes permettent de le faire sur une large gamme de valeurs (de "RTT_MIN" à "RTT_MAX") de GRTT tout en conservant une gamme supérieure de précision pour les

petites valeurs et moins de précision pour les grandes valeurs. Les valeurs de 0,000001 à 0,6 secondes et de 1000 secondes sont RECOMMANDÉES pour respectivement "RTT_MIN" et "RTT_MAX". Les applications de diffusion groupée fiable fondée sur le NACK peuvent souhaiter placer une limite supérieure supplémentaire plus petite sur le GRTT annoncé par les envoyeurs pour satisfaire des contraintes de latence de livraison des données d'application au prix d'un volume supérieur de rétroactions dans certains environnements de réseau.

```
unsigned char QuantizeGrtt(double grtt)
{
   si (grtt > RTT_MAX)
      grtt = RTT_MAX;
   autrement si (grtt < RTT_MIN)
      grtt = RTT_MIN;
   si (grtt < (33*RTT_MIN))
      retourner ((unsigned char)(grtt / RTT_MIN) - 1);
   autrement
      retourner ((unsigned char)(ceil(255,0 – (13,0 * log(RTT_MAX/grtt)))));
}
double UnquantizeRtt(unsigned char qrtt)
{
   retourner ((qrtt ≤ 31) ?
         (((double)(qrtt+1))*(double)RTT_MIN) :
         (RTT_MAX/exp(((double)(255-qrtt))/(double)13,0)));
}
```

Noter que cette fonction est utile pour quantifier les temps de GRTT dans la gamme de 1 microseconde à 1000 secondes. Bien sûr, les mises en œuvre de protocole de diffusion groupée fiable fondée sur le NACK peuvent souhaiter contraindre plus les estimations annoncées de GRTT (par exemple, limiter la valeur maximum) pour des raisons pratiques.

3.8 Détermination/estimation de taille de groupe

Quand le fonctionnement du protocole de diffusion groupée fiable fondée sur le NACK inclut des mécanismes qui excitent des rétroactions de la part du groupe au sens large (par exemple, contrôle d'encombrement) il est possible d'estimer grossièrement la taille de groupe sur la base du nombre de messages de rétroaction reçus par rapport à la distribution du mécanisme probabiliste de suppression utilisé. Noter que le mécanisme de suppression fondé sur le temporisateur décrit dans ce document n'exige pas une estimation très précise de l'estimation de la taille de groupe pour fonctionner de façon adéquate. Donc, une estimation grossière, en particulier si elle est gérée de façon prudente, peut suffire. L'estimation de la taille de groupe peut aussi être déterminée administrativement. En l'absence de tout mécanisme de détermination d'estimation de la taille de groupe, une estimation par défaut de la valeur de la taille de groupe de 10 000 est RECOMMANDÉE pour une gestion raisonnable des rétroactions étant donnée l'adaptabilité de l'usage attendu de la diffusion groupée fiable fondée sur le NACK. Cette estimation prudente (sur-estimée) de l'estimation de la taille de groupe dans les algorithmes décrits ci-dessus va résulter en une latence accrue du processus de réparation de NACK si la taille réelle du groupe est plus petite mais avec une garantie de protection contre l'explosion de rétroactions. L'étude du mécanisme de suppression de retour fondé sur le temporisateur décrite dans [McastFeedback] et [NormFeedback] a montré que l'estimation de la taille de groupe a seulement besoin d'être d'un ordre de grandeur pour fournir des performances de suppression efficaces.

3.9 Fonctionnement du contrôle d'encombrement

Le contrôle d'encombrement qui partage équitablement la capacité réseau disponible avec d'autres instances de diffusion groupée fiable et TCP est EXIGÉ pour le fonctionnement général de l'Internet. Les techniques de contrôle d'encombrement de diffusion groupée convivial sur TCP (TFMCC, *TCP-Friendly Multicast Congestion Control*) [TfmccPaper] ou le contrôle d'encombrement pragamatique général de diffusion groupée (PGMCC, *Pragmatic General Multicast Congestion Control*) [PgmccPaper] peuvent être appliquées au fonctionnement de la diffusion groupée fiable fondée sur le NACK pour satisfaire cette exigence. La première de ces technique a été aussi documentée dans la [RFC4654] et a été appliquée avec succès dans le protocole de diffusion groupée fiable fondée sur le NACK (NORM, *NACK-Oriented Reliable Multicast Protocol*) [RFC3940].

3.10 Assistance de système intermédiaire

Les protocoles de diffusion groupée fondée sur le NACK peuvent bénéficier de l'assistance de systèmes intermédiaires d'utilité générale. En particulier, une suppression de NACK supplémentaire où les systèmes intermédiaires peuvent agréger le contenu de NACK (ou filtrer les contenus de NACK dupliqués) provenant des receveurs lorsque il est relayé vers l'envoyeur pourraient améliorer l'adaptabilité de la taille de groupe de NORM. Pour les protocoles de diffusion groupée fiable fondée sur le NACK qui utilisent la FEC, il est possible que les systèmes intermédiaires puissent être capables de filtrer les messages de réparation de FEC pour fournir une "sous invocation" intelligente de contenu de réparation aux différentes branches de la topologie de diffusion groupée selon les besoins de réparation appris des précédents NACK de receveurs. De même, les systèmes intermédiaires pourraient surveiller les NACK de receveurs et fournir des transmissions de réparation à la demande en réponse si de l'état suffisant sur le contenu transmis a été conservé. Cela peut réduire la latence et le volume des transmissions de réparation quand le système intermédiaire est associé à une liaison de réseau qui est particulièrement problématique à l'égard de la perte de paquets. Ces types de fonctions d'assistance exigeraient du système intermédiaire une interprétation des identifiants et fanions de contenu d'unité de données de transport. Les conceptions de protocole fondé sur le NACK devraient considérer le potentiel d'assistance des systèmes intermédiaires dans la spécification des messages et opérations de protocole. Il est probable que l'assistance des systèmes intermédiaires va être plus pragmatique si les exigences d'analyse de message sont modestes et si la quantité d'état qu'il est exigé que conserve un système intermédiaire est relativement petite.

4. Applicabilité de la diffusion groupée fiable fondée sur le NACK

Le bloc de construction de NACK de diffusion groupée s'applique aux protocoles qui souhaitent employer des accusés de réception négatifs pour réaliser un transfert de données fiable. Les protocoles de diffusion groupée fiable fondée sur le NACK conçus de façon appropriée offrent des avantages d'adaptabilité pour les applications et/ou topologies de réseau où, pour diverses raisons, il est prohibitif de construire une infrastructure de livraison d'ordre supérieur au dessus du service IP de diffusion groupée de base de couche 3 (par exemple, des arborescences de distribution de données en envoi individuel ou hybrides envoi individuel/diffusion groupée). De plus, la propriété d'adaptabilité à la diffusion groupée des protocoles fondés sur le NACK [RmComparison], [RmClasses] est applicable lorsque une large "ventilation" est attendue pour un seul bond de réseau (par exemple, livraison de données de télévision par câble, par satellite, ou autres services de communication en diffusion). De plus, la simplicité d'un protocole fondé sur une distribution "plate" de diffusion groupée à l'échelle d'un groupe peut offrir des avantages pour une large gamme de services distribués ou des réseaux ou applications dynamiques. Les protocoles de diffusion groupée fiable fondée sur le NACK peuvent faire usage de communication en diffusion groupée réciproque (entre envoyeurs et receveurs) sous le modèle de diffusion groupée toutes sources (ASM, *Any-Source Multicast*) défini dans la [RFC1112], et sont capables d'un fonctionnement adaptable dans des topologies asymétriques, comme la diffusion groupe spécifique de source (SSM, *Source-Specific Multicast*) [RFC4607], où il peut seulement y avoir un service d'acheminement en envoi individuel des receveurs à l'envoyeur.

Le fonctionnement du protocole de diffusion groupée fiable fondée sur le NACK est compatible avec les techniques de codage de correction d'erreur directe de couche transport décrites dans la [RFC3453] et les mécanismes de contrôle d'encombrement comme ceux décrits dans [TfmccPaper] et [PgmccPaper]. La limitation principale du fonctionnement de la diffusion groupée fiable fondée sur le NACK implique l'adaptabilité de la taille de groupe quand la capacité du réseau pour les rétroactions du receveur est très limitée. Il est possible que, avec une conception appropriée du protocole, les techniques d'assistance de système intermédiaire mentionnées au paragraphe 2.4 et décrites plus en détails au paragraphe 3.10 puissent permettre des approches fondées sur le NACK qui s'adaptent à de plus grandes tailles de groupes. Le fonctionnement de la diffusion groupée fiable fondée sur le NACK est aussi gouverné par les contraintes de mise en mémoire tampon des mises en œuvre. Une mise en mémoire tampon supérieure à celle exigée pour le transport fiable normal en point à point (par exemple, TCP) est recommandée pour permettre des disparités dans la connexité du groupe receveur et permettre que les délais de rétroaction exigés pour atteindre l'adaptabilité de la taille de groupe.

Les travaux expérimentaux antérieurs incluaient diverses instances de protocole qui mettaient en œuvre certains des concepts décrit dans le présent document de blocs de construction. Cela inclut le protocole de diffusion groupée pragmatique générale (PGM, *Pragmatic General Multicast*) décrit dans la [RFC3208] ainsi que d'autres qui ont été documentés ou déployés en-dehors des activités de l'IETF. Alors que la spécification du protocole PGM et quelques autres approches ont englobé beaucoup des objectifs de la livraison de données en vrac comme décrit ici, celle fondée sur le bloc de construction de NACK fournit un cadre plus général afin que des besoins d'application différents puissent être satisfaits par différentes variantes d'instances de protocole. L'approche du bloc de construction fondé sur le NACK décrite ici inclut la compatibilité avec les autres mécanismes de protocole incluant la FEC et le contrôle d'encombrement qui sont décrits dans d'autres documents de bloc de construction de diffusion groupée fiable de l'IETF. Le processus de réparation de NACK décrit dans ce document peut fournir des avantages de performances comparés à PGM quand les deux sont

déployés sur une pure base de bout en bout sans assistance de système intermédiaire. L'estimation du temps d'aller-retour décrite ici et son utilisation dans le processus de réparation de NACK permet un fonctionnement du protocole qui s'adapte plus automatiquement aux différents environnements de réseau ou fonctionne dans des environnements où la connexité est dynamique. L'utilisation des techniques d'identification de charge utile de FEC décrite dans le bloc de construction de FEC [RFC5052] et les instances spécifiques de FEC permettent aux instances de protocole plus de souplesse lorsque les techniques de FEC évoluent que le schéma d'identification de données spécifiques de numéro de séquence décrit dans la spécification de PGM. Une souplesse similaire est attendue si les instances de protocole sont conçues pour invoquer de façon modulaire (au moment de la conception sinon au démarrage) le bloc de construction approprié de contrôle d'encombrement pour les différentes applications ou déploiements.

5. Considérations sur la sécurité

Les protocoles de diffusion groupée fiable fondée sur le NACK sont supposés être sujets aux mêmes vulnérabilités de sécurité que les autres protocoles IP de diffusion groupée. Cependant, à la différence des protocoles de transport en point à point (en envoi individuel) il est possible qu'un participant au mauvais comportement puisse impacter le ressenti de service de transport des autres membres du groupe. Par exemple, un nœud receveur malveillant pourrait intentionnellement transmettre des messages de NACK pour causer la transmission par l'envoyeur de réparations de transmission inutiles au lieu de faire des progrès avec le transfert fiable. Aussi, des messages au niveau du groupe pour prendre en charge le contrôle d'encombrement ou d'autres aspects du fonctionnement du protocole peuvent être sujets à des vulnérabilités similaires. Donc, il est fortement RECOMMANDÉ que des techniques de sécurité comme l'authentification et la vérification de l'intégrité des données soient appliquées pour les déploiements de diffusion groupée fiable fondée sur le NACK. Les instances de protocole qui utilisent ce bloc de construction DOIVENT identifier les approches de sécurité qui peuvent être utilisées pour traiter ces considérations de sécurité et les autres.

La diffusion groupée fiable fondée sur le NACK est compatible avec les mécanismes d'authentification de la sécurité IP (IPsec) [RFC4301] qui sont RECOMMANDÉS pour la protection contre l'intrusion de session et les attaques de déni de service. Une menace particulière pour les protocoles fondés sur le NACK est celle des attaques en répétition de NACK, qui pourraient empêcher un envoyeur de diffusion groupée de faire des progrès dans la transmission. Tous les mécanismes standard de IPsec qui peuvent fournir une protection contre de telles attaques en répétition sont RECOMMANDÉS. Le groupe de travail Sécurité de la diffusion groupée (MSEC) de l'IETF a développé un ensemble de recommandations dans "Extensions de diffusion groupée à l'architecture de sécurité pour le protocole Internet" [RFC5374] qui peuvent être appliquées pour étendre de façon appropriée les mécanismes IPsec au fonctionnement en diffusion groupée. Un appendice de ce document traite spécifiquement du modèle de service de protocole de diffusion groupée fiable en mode NACK. Comme la prise en charge complète du fonctionnement en diffusion groupée pour IPsec peut suivre éventuellement le déploiement de la diffusion groupée fiable, les instances de protocole de diffusion groupée fiable fondée sur le NACK DEVRAIENT envisager de fournir la prise en charge de leur propre protection contre l'attaque en répétition de NACK quand les mécanismes de couche réseau ne sont pas disponibles. Cela PEUT être nécessaire lorsque des mises en œuvre de IPsec sont utilisées qui ne fournissent pas de protection contre l'attaque en répétition de diffusion groupée quand plusieurs sources sont présentes.

Pour les déploiement de diffusion groupée fondée sur le NACK avec de grands groupes de receveurs qui utilisent IPsec, des approches utilisant des clés partagées communes pourraient être développées pour les messages de protocole générés par le receveur pour tenir un nombre pratique d'associations de sécurité IPsec. Cependant, une telle authentification fondée sur le groupe peut n'être pas suffisante si la population de receveurs ne peut être complètement de confiance. De plus, cela peut rendre l'identification de nœuds receveur au mauvais comportement (bien qu'authentifiés) problématique car des nœuds pourraient se faire passer pour d'autres receveurs du groupe. Dans des déploiements comme ceux-là, on DEVRAIT envisager l'utilisation de la diffusion groupée spécifique de source (SSM, *source-specific multicast*) au lieu des modèles de diffusion groupée toute source (ASM, *Any-Source Multicast*). Le fonctionnement en SSM peut simplifier les défis de sécurité de deux façons :

1. Un protocole fondé sur le NACK qui prend en charge le fonctionnement en SSM peut éliminer la signalisation directe de receveur à receveur. Cela réduit dramatiquement le nombre d'associations de sécurité à établir.

2. Le ou les envoyeurs de SSM peuvent fournir un point de gestion centralisé pour le fonctionnement sûr de groupe pour leurs flux de données respectifs car il est exigé de l'envoyeur seul de conduire l'authentification individuelle des hôtes pour chaque receveur quand l'authentification fondée sur le groupe ne suffit pas ou n'est pas pratique à déployer.

Quand l'authentification individuelle des hôtes est exigée, il est alors possible que les receveurs utilisent une signature numérique sur la charge utile du protocole d'encapsulation de charge utile de sécurité IPsec (ESP) comme décrit dans la

[RFC4359]. Un système de signature fondé sur l'identité ou une infrastructure de clé publique spécifique de groupe pourrait éviter un état par receveur chez le ou les envoyeurs. De plus, les mises en œuvre DOIVENT aussi prendre en charge des politiques de limitation de l'impact de receveurs extrêmement ou exceptionnellement mauvais (dû à un mauvais comportement ou autrement) sur le fonctionnement global du groupe si c'est acceptable pour l'application concernée.

Comme décrit au paragraphe 3.4, le déploiement de la diffusion groupée fiable fondée sur le NACK dans certains environnements de réseau peut exiger l'identification des membres du groupe au delà de celle de l'adressage IP. Si des mécanismes de sécurité spécifiques du protocole sont développés, alors il est RECOMMANDÉ que des identifiants de membre de groupe du protocole soient utilisés comme sélecteurs (comme défini dans la [RFC4301]) pour les associations de sécurité applicables. Quand IPsec est utilisé, il est RECOMMANDÉ que les mises en œuvre de protocole vérifient que les adresses IP de source des paquets reçus sont valides pour l'identifiant de source du protocole donné en plus de l'authentification IPsec usuelle. Cela va empêcher un membre au mauvais comportement (bien qu'autorisé) de falsifier des messages provenant d'autres membres légitimes, pourvu que l'authentification individuelle des hôtes soit prise en charge.

Le groupe de travail MSEC a aussi développé des solutions de chiffrement automatique de groupe qui sont applicables à la sécurité de la diffusion groupée fiable fondée sur le NACK. Par exemple, pour prendre en charge IPsec ou d'autres mécanismes de sécurité, le protocole de gestion de clé d'association de groupe sécurisé [RFC4535] PEUT être utilisé pour la gestion automatique de clé de groupe. La technique qu'il identifie pour "l'établissement de groupe pour les membres en réception seule" peut être le fonctionnement SSM d'une application de diffusion groupée fiable fondée sur le NACK.

6. Changements par rapport à la RFC 3941

Cette section donne la liste des changements entre la version expérimentale de cette spécification, la [RFC3941], et la présente version :

1. Changement du titre pour éviter la confusion avec la spécification du protocole NORM.
2. Mise à jour des références, mise à jour des documents de bloc de construction de RMT, et
3. Considérations sur la sécurité plus détaillées.

7. Remerciements

(et ils ne sont pas négatifs)

Les auteurs tiennent à remercier George Gross, Rick Jones, et Joerg Widmer de leurs précieux commentaires sur ce document. Ils tiennent aussi à remercier les présidents du groupe de travail RMT, Roger Kermode et Lorenzo Vicisano, de leur soutien au développement de cette spécification, et Sally Floyd pour ses apports précoces au document.

8. Références

8.1 Références normatives

[RFC1112] S. Deering, "Extensions d'hôte pour diffusion groupée sur IP", STD 5, août 1989. *(MàJ par* RFC2236*)*

[RFC2119] S. Bradner, "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997. *(MàJ par* RFC8174)

[RFC4607] H. Holbrook, B. Cain, "Diffusion groupée spécifique de source pour IP", août 2006. *(P.S.)*

8.2 Références pour information

[ArchConsid.] Clark, D. and D. Tennenhouse, "Architectural Considerations for a New Generation of Protocols", Proc. ACM SIGCOMM, pp. 201-208, septembre 1990.

[DelayEstimation] Ozdemir, V., Muthukrishnan, S., and I. Rhee, "Scalable, Low-Overhead Network Delay Estimation", NCSU/AT&T White Paper, février 1999.

[FecBroadcast] Metzner, J., "An Improved Broadcast Retransmission Protocol", IEEE Transactions on Communications Vol. Com-32, No. 6, juin 1984.

[FecHybrid] Gossink, D. et J. Macker, "Reliable Multicast and Integrated Parity Retransmission with Channel Estimation", IEEE Globecomm 1998, 1998.

[McastFeedback] Nonnenmacher, J. and E. Biersack, "Optimal Multicast Feedback", IEEE Infocom p. 964, mars/avril 1998.

[NormFeedback] Adamson, B. and J. Macker, "Quantitative Prediction of NACK-Oriented Reliable Multicast (NORM) Feedback", IEEE MILCOM 2002, octobre 2002.

[PgmccPaper] Rizzo, L., "pgmcc: A TCP-Friendly Single-Rate Multicast Congestion Control Scheme", ACM SIGCOMM 2000, août 2000.

[RFC2357] A. Mankin, A. Romanov, S. Bradner et V. Paxson, "Critères de l'IETF pour l'évaluation des protocoles de transport et d'application de diffusion groupée fiable", juin 1998. *(Information)*

[RFC3208] T. Speakman et autres, "Spécification du protocole PGM de transport fiable", décembre 2001. *(Expérimentale)*

[RFC3269] R. Kermode et L. Vicisano, "Lignes directrices pour les auteurs de documents de mise en œuvre de protocole et de blocs de construction de transport fiable en diffusion groupée (RMT)", avril 2002.

[RFC3453] M. Luby et autres, "Utilisation de la correction d'erreur directe (FEC) en diffusion groupée fiable", décembre 2002. *(Info.)*

[RFC3940] B. Adamson et autres, "Protocole de diffusion groupée fiable orientée (NORM) accusé de réception négatif (NACK)", novembre 2004. *(Expérimentale ; Remplacée par* RFC5740)

[RFC3941] B. Adamson et autres, "Blocs de construction de diffusion groupée fiable orientée (NORM) accusé de réception négatif (NACK)", novembre 2004. *(Expérimentale ; obsolète, voir* RFC5401)

[RFC4301] S. Kent et K. Seo, "Architecture de sécurité pour le protocole Internet", décembre 2005. *(P.S.) (Remplace la* RFC2401)

[RFC4359] B. Weis, "Utilisation des signatures RSA/SHA-1 au sein d'une charge utile de sécurité par encapsulation (ESP) et d'un en-tête d'authentification (AH)", janvier 2006. *(P.S.)*

[RFC4535] H. Harney et autres, "GSAKMP : protocole de gestion de clés d'association de groupe sécurisé", juin 2006. *(P.S.)*

[RFC4654] J. Widmer, M. Handley, "Spécification du protocole de contrôle d'encombrement de diffusion groupée compatible TCP (TFMCC)", août 2006. *(Expérimentale)*

[RFC5052] M. Watson et autres, "Bloc de construction de la correction d'erreur directe (FEC)", août 2007. (*Remplace* RFC3452) *(P.S.)*

[RFC5374] B. Weis et autres, "Extensions de diffusion groupée à l'architecture de sécurité du protocole Internet", novembre 2008. (*P.S.)*

[RFC5445] M. Watson, "Schémas de base de correction d'erreur directe (FEC)", mars 2009. (*Remplace* RFC3452, RFC3695) *(P. S.)*

[RFC5510] J. Lacan, V. Roca, J. Peltotalo, S. Peltotalo, "Schémas Reed-Solomon de correction d'erreur directe (FEC)", avril 2009. *(P.S.)*

[RmClasses] Levine, B. and J. Garcia-Luna-Aceves, "A Comparison of Known Classes of Reliable Multicast Protocols", Proc. International Conference on Network Protocols (ICNP- 96) Columbus, OH, octobre 1996.

[RmComparison] Pingali, S., Towsley, D., and J. Kurose, "A Comparison of Sender-Initiated and Receiver-Initiated Reliable Multicast Protocols", Proc. INFOCOMM San Francisco, CA, octobre 1993.

[RmFec] Macker, J., "Reliable Multicast Transport and Integrated Erasure-based Forward Error Correction", IEEE MILCOM 1997, octobre 1997.

[SrmFramework] Floyd, S., Jacobson, V., McCanne, S., Liu, C., and L. Zhang, "A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing", Proc. ACM SIGCOMM, août 1995.

[TfmccPaper] Widmer, J. et M. Handley, "Extending Equation-Based Congestion Control to Multicast Applications", ACM SIGCOMM 2001, août 2001.

Adresse des auteurs

Brian Adamson
Naval Research Laboratory
Washington, DC 20375
mél : adamson@itd.nrl.navy.mil

Carsten Bormann
Universitaet Bremen TZI
Postfach 330440
D-28334 Bremen, Germany
mél : cabo@tzi.org

Mark Handley
University College London
Gower Street
London, WC1E 6BT UK
mél : M.Handley@cs.ucl.ac.uk

Joe Macker
Naval Research Laboratory
Washington, DC 20375
mél : macker@itd.nrl.navy.mil