

Groupe de travail Réseau	C. Perkins ; I. Kouvelas ; O. Hodson ; V. Hardman,
Request for Comments : 2198	University College London
Catégorie : En cours de normalisation	M. Handley, ISI
	J.C. Bolot ; A. Vega-Garcia ; S. Fosse-Parisis, INRIA Sophia Antipolis
Traduction Claude Brière de L'Isle	septembre 1997

Charge utile RTP pour données audio redondantes

Statut du présent mémoire

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

Résumé

Le présent document décrit un format de charge utile à utiliser avec le protocole de transport en temps réel (RTP) version 2, pour le codage des données audio redondantes. La principale motivation du schéma décrit ici est le développement d'outils d'audio conférence à utiliser sur des réseaux qui subissent des pertes de paquets tels que le Mbone Internet, bien que ce schéma ne soit pas limité à de telles applications.

1 Introduction

Si la conférence multimédia doit devenir largement utilisée par la communauté Mbone de l'Internet, les utilisateurs doivent percevoir une qualité suffisante pour la plupart des applications. Un certain nombre de problèmes qui impactent la qualité des conférences ont été identifiés, dont le plus significatif est la perte de paquet. C'est un problème persistant, en particulier dans le contexte de popularité croissante, et donc de charge croissante, de l'Internet. Les perturbations de l'intelligibilité de la parole, même à de faibles taux de pertes que l'on rencontre actuellement peuvent convaincre toute une génération d'utilisateurs que la conférence multimédia sur l'Internet n'est pas viable. L'ajout de redondances dans le flux des données est proposé comme solution [1]. Si un paquet est perdu, les informations manquantes peuvent être reconstruites chez le receveur à partir des données redondantes qui arrivent dans le ou les paquets suivants, pourvu que le nombre moyen de paquets perdus consécutifs reste faible. Des travaux récents, [4], [5], montrent que les schémas de perte de paquet dans l'Internet sont tels que ce schéma fonctionne normalement bien.

Le présent document décrit un format de charge utile RTP pour la transmission de données audio codées de cette façon redondante. La section 2 présente les exigences et les motivations qui conduisent à la définition de ce format de charge utile, et ne fait pas partie de la définition du format de charge utile. Les sections 3 et suivantes définissent le format de charge utile RTP pour les données audio redondantes.

2 Exigences/motivations

Les exigences d'un schéma de codage redondant sous RTP sont les suivantes :

- o Les paquets doivent porter un codage primaire et un ou plusieurs codages redondants.
- o Comme une multitude de codages peuvent être utilisés pour les informations redondantes, chaque bloc de codage redondant doit avoir un identifiant de type de codage.
- o Comme il est souhaitable d'utiliser des codages de taille variable, chaque bloc codé dans le paquet doit avoir un indicateur de longueur.
- o L'en-tête RTP fournit un champ d'horodatage qui correspond à l'heure de création des données codées. Lorsque des codages redondants sont utilisés, ce champ d'horodatage peut se référer à l'heure de création des données du codage primaire. Les blocs de données redondants vont correspondre aux différents intervalles de temps avec les données primaires, et donc chaque bloc de codage redondant va exiger son propre horodatage. Pour réduire le nombre d'octets nécessaires pour porter l'horodatage, il peut être codé comme différence de l'horodatage du codage redondant et l'horodatage du primaire.

Il y a essentiellement deux moyens pour ajouter les données audio redondantes à la spécification RTP standard : une extension d'en-tête peut contenir la redondance, ou un ou plusieurs types supplémentaires de charge utile peuvent être définis.

Inclure toutes les informations de redondance pour un paquet dans une extension d'en-tête rendrait facile aux applications qui ne mettent pas en œuvre la redondance de l'éliminer et de ne traiter que les données du codage primaire. Il y a cependant un certain nombre d'inconvénients à ce schéma :

- o Il y a une grosse redondance à partir du nombre d'octets nécessaires pour l'en-tête d'extension (4) et le possible bourrage qui est nécessaire à la fin de l'extension pour arrondir à une limite de quatre octets (jusqu'à trois octets). Pour de nombreuses applications, cette redondance est inacceptable.
- o L'utilisation de l'extension d'en-tête limite les applications à un seul codage redondant, sauf si une autre structure est introduite dans l'extension. Il en résulterait encore plus de redondance.

Pour ces raisons, l'utilisation de l'extension d'en-tête RTP pour contenir les codages audio redondants est déconseillée.

Le profil RTP pour les conférences audio et vidéo [3] énumère un ensemble de types de charges utiles et donne une gamme dynamique de 32 codages qui peuvent être définis à travers un protocole de contrôle de conférence. Cela conduit à deux schémas possibles pour allouer des types de charge utile RTP supplémentaires pour les applications de redondance audio :

1. Un schéma de codage dynamique peut être défini, pour chaque combinaison de types de charge utile primaire/redondante, en utilisant la gamme de types de charge utile RTP dynamique.
2. Un seul type fixe de charge utile peut être défini comme représentant un paquet avec redondance. Il peut ensuite être alloué soit à un type de charge utile RTP statique, soit à un type de charge utile qui sera alloué de façon dynamique.

Il est possible de définir un ensemble de types de charge utile qui signifie une combinaison particulière de codages primaires et secondaires pour chacun des 32 types de charges utiles dynamiques fournis. Cela serait une solution légèrement restrictive mais faisable pour les paquets ayant un seul bloc de redondance car le nombre de combinaisons possibles n'est pas très grand. Cependant, le besoin de plusieurs blocs de redondance augmente grandement le nombre de combinaisons de codage et rend cette solution non viable.

Une version modifiée de la solution ci-dessus pourrait être de décider, avant de commencer une conférence, d'un ensemble de 32 combinaisons de codages qui seraient utilisés pour la durée de la conférence. Tous les outils de la conférence peuvent être initialisés avec cet ensemble actif de combinaisons de codage. La communication de l'ensemble actif pourrait être faite en utilisant un mécanisme externe, hors bande. L'établissement est compliqué car il faut prendre grand soin que les outils de démarrage aient des paramètres identiques. Ce schéma est plus efficace car un seul octet est utilisé pour identifier les combinaisons de codages.

On estime que la complication inhérente à la distribution de la transposition des types de charge utile en combinaisons de données redondantes empêche d'utiliser ce mécanisme.

Une solution plus souple est d'avoir un seul type de charge utile qui signifie un paquet avec redondance. Ce paquet devient alors un conteneur, encapsulant plusieurs charges utiles dans un seul paquet RTP. Un tel schéma est flexible, car toute quantité de redondance peut être encapsulée au sein d'un seul paquet. Il y a cependant une petite redondance dans la mesure où chaque charge utile encapsulée doit être précédée d'un en-tête qui indique le type de données incluses. C'est la solution préférée, car elle est à la fois souple, extensible, et a une redondance relativement faible. Le reste de ce document est consacré à la description de cette solution.

3 Spécification du format de charge utile

Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" dans ce document sont à interpréter comme décrit dans la RFC2119 [7].

L'affectation d'un type de charge utile RTP à ce nouveau format de paquet sort du domaine d'application du présent document, et ne sera pas spécifié ici. On s'attend à ce que le profil RTP pour une classe particulière d'applications affecte un type de charge utile pour ce codage, ou si ce n'est pas fait, qu'un type de charge utile soit choisi dans la gamme dynamique.

Un paquet RTP qui contient des données redondantes devra avoir un en-tête RTP standard, avec un type de charge utile

indiquant la redondance. Les autres champs de l'en-tête RTP se rapportent au bloc de données principal des données redondantes.

À la suite de l'en-tête RTP se trouvent un certain nombre d'en-têtes supplémentaires, définis dans la figure ci-dessous, qui spécifient le contenu de chacun des codages portés par le paquet. À la suite de ces en-têtes supplémentaires se trouvent un certain nombre de blocs de données qui contiennent les données de charge utile RTP standard pour ces codages. On note que tous les en-têtes sont verrouillés sur une limite de 32 bits, mais que les données de charge utile ne sont normalement pas verrouillées. Si plusieurs codages redondants sont portés dans un paquet, ils devraient correspondre à des intervalles de temps différents : il n'y a pas de raison d'inclure plusieurs copies des données d'un seul intervalle de temps dans un paquet.

```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|F| TCU du bloc | Décalage d'horodatage | Longueur du bloc |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Les bits dans l'en-tête sont spécifiés comme suit :

F : Le premier bit (*First*) de l'en-tête indique si un autre bloc d'en-tête suit. 1 indique qu'un ou d'autres blocs suivent, 0 indique qu'il s'agit du dernier bloc d'en-tête.

TCU du bloc : ces 7 bits indiquent le type de charge utile RTP pour ce bloc.

Décalage d'horodatage : c'est un entier non signé sur 14 bits qui indique le décalage de l'horodatage de ce bloc par rapport à l'horodatage donné dans l'en-tête RTP. L'utilisation d'un décalage non signé implique que les données redondantes doivent être envoyées après des données principales, et que c'est donc un temps à soustraire de l'horodatage actuel pour déterminer l'horodatage des données dont ce bloc est la redondance.

Longueur de bloc : ces 10 bits indiquent la longueur en octets du bloc de données correspondant, en-tête exclu.

On notera que l'utilisation d'un décalage d'horodatage en valeur absolue limite légèrement l'utilisation des données redondantes : il n'est pas possible d'envoyer la redondance avant le codage primaire. Cela peut affecter des schémas où un codage à faible bande passante convenable pour la redondance est produit de façon précoce dans le processus de codage, et qu'il serait donc faisable de le transmettre plus tôt. Cependant, l'ajout du signe réduirait de façon inacceptable la gamme de valeurs des décalages d'horodatage, et l'augmentation de la taille du champ au delà de 14 bits limiterait le champ de longueur du bloc. Il semble que limiter la redondance à être transmise après le primaire posera moins de problèmes que de limiter la taille des autres champs.

Le décalage de l'horodatage d'un bloc redondant est mesuré dans les mêmes unités que l'horodatage du codage primaire (c'est-à-dire que les échantillons audio ont le même débit d'horloge que ceux du primaire). Cela implique que le codage redondant DOIT être échantillonné au même taux que le primaire.

On notera de plus que la longueur de bloc et le décalage d'horodatage sont respectivement de 10 bits et 14 bits; plutôt que des 8 et 16 bits qui auraient pu paraître évidents. Bien qu'un tel codage complique légèrement l'analyse des informations de l'en-tête, et ajoute un peu de redondance de traitement supplémentaire, le choix le plus évident pose un certain nombre de problèmes: Un champ de longueur de bloc de 8 bits est suffisant pour la plupart des codages possibles, mais pas tous : par exemple les paquets MIC de 80 ms et audio DVI (*interfac vidéo numérique*) comportent plus de 256 octets, et ne peuvent pas être codés avec un champ de longueur d'un seul octet. Il est possible d'imposer une structure supplémentaire au champ Longueur de bloc (par exemple le bit de poids fort à un pourrait impliquer que les 7 bits de moindre poids codent une longueur en mots, plutôt qu'en octets) cependant, de tels schémas sont complexes. L'utilisation d'un champ Longueur de bloc de 10 bits conserve la simplicité et fournit une gamme élargie, au prix d'une gamme réduite de valeurs d'horodatage.

L'en-tête du bloc de codage primaire est placé en dernier dans le paquet. Il est donc possible d'omettre les champs Horodatage et Longueur de bloc de l'en-tête de ce bloc, car ils peuvent être déterminés à partir de l'en-tête RTP et de la longueur globale du paquet. L'en-tête pour le bloc primaire (final) comporte seulement un bit F à zéro, et les informations de type de charge utile du bloc, pour un total de 8 bits. C'est illustré par la figure ci-dessous :

```

0 1 2 3 4 5 6 7
+---+---+---+---+---+---+
|0| TCU du bloc |
+---+---+---+---+---+---+

```

L'en-tête final est suivi immédiatement par les blocs de données, mémorisés dans le même ordre que les en-têtes. Il n'y a

pas de bourrage ou autre délimiteur entre les blocs de données, et ils ne sont normalement pas verrouillés sur 32 bits. Là encore, ce choix a été fait pour réduire la consommation de bande passante, aux dépens du temps de décodage supplémentaire.

Le choix des codages utilisés devrait refléter les exigences de bande passante de ces codages. Il est prévu que le codage redondant devra utiliser nettement moins de bande passante que le codage primaire ; l'exception étant le cas où le primaire a très peu de bande passante et de fortes exigences de traitement, auquel cas, une copie du primaire PEUT être utilisée comme redondance. Le codage redondant NE DOIT PAS consommer plus de bande passante que le primaire.

L'utilisation de plusieurs niveaux de redondance est rarement nécessaire. Cependant, dans les cas qui l'exigent, la bande passante requise par chaque niveau de redondance est supposée être significativement moindre que celle du précédent niveau.

4 Limitations

Le bit marqueur RTP n'est pas préservé pour les blocs de données en redondance. Donc, si le primaire (qui contient ce marqueur) est perdu, le marqueur est perdu. On pense que cela ne causera pas de problèmes : même si le bit marqueur était transmis avec les informations redondantes, il y aurait encore la possibilité qu'il soit perdu, de sorte que les applications devraient quand même garder l'idée de cette possibilité.

De plus, les informations de CSRC ne sont pas préservées pour les données redondantes. Les données de CSRC dans l'en-tête RTP d'un paquet audio redondant ne se rapportent qu'au primaire. Comme les données de CSRC dans un flux audio sont supposées ne changer que relativement rarement, il est recommandé que les applications qui exigent ces informations supposent que les données de CSRC dans l'en-tête RTP puissent être appliquées aux données redondantes reconstruites.

5 Relation avec SDP

Lorsque on utilise une charge utile redondante, elle peut devoir être liée à un type de charge utile RTP dynamique. Cela peut se faire par n'importe quel mécanisme hors bande, mais une façon courante de le faire est de communiquer cette liaison en utilisant le protocole de description de session (SDP, *Session Description Protocol*) [6]. SDP a un mécanisme pour lier des types de charge utile dynamiques à des codecs particuliers, des taux d'échantillon, et un certain nombre de canaux en utilisant l'attribut "rtptime". Un exemple de cette utilisation (avec le profil audio/vidéo RTP [3]) est :

```
m=audio 12345 RTP/AVP 121 0 5
a=rtptime:121 red/8000/1
```

Cela spécifie qu'un flux audio qui utilise RTP se sert des types de charge utile 121 (un type de charge utile dynamique), 0 (MIC loi μ) et 5 (DVI). L'attribut "rtptime" est utilisé pour lier le type de charge utile 121 au codec "red" ce qui indique que ce codec est en fait une trame de redondance, 8 kHz, et monaural. Lorsque il est utilisé avec SDP, le terme "red" sert à indiquer le format de redondance décrit dans le présent document.

Dans ce cas, les formats supplémentaires de MIC et DVI sont spécifiés. Le receveur doit donc être prêt à utiliser ces formats. Une telle spécification signifie que l'envoyeur enverra la redondance par défaut, mais peut aussi envoyer du MIC ou du DVI. Cependant, avec une charge utile redondante on doit comprendre que cela signifie qu'aucun codec autre que MIC ou DVI ne sera utilisé dans les codages redondants. Noter que les formats de charge utile supplémentaires définis dans le champ "m=" peuvent eux-mêmes être des types de charge utile dynamiques, et s'il en est ainsi, un certain nombre d'attributs "a=" supplémentaires peuvent être nécessaires pour décrire ces types de charge utile dynamiques.

Pour recevoir un flux redondant, c'est tout ce qui est nécessaire. Cependant, pour envoyer un flux redondant, l'envoyeur a besoin de savoir quels codecs sont recommandés pour les codages primaire et secondaire (et tertiaire, etc). Ces informations sont spécifiques du format de redondance, et sont spécifiées en utilisant un attribut "fmtp" supplémentaire qui porte des informations spécifiques du format. Un répertoire de session n'analyse pas les valeurs spécifiées dans un attribut fmtp mais les passe simplement inchangées à l'outil de support. Pour la redondance, on définit les paramètres de format comme étant une liste de types de charge utile RTP séparés par des barres obliques "/".

Un exemple complet est donc :

```
m=audio 12345 RTP/AVP 121 0 5
a=rtptime:121 red/8000/1
a=fmtp:121 0/5
```

Cela spécifie que le format par défaut pour les envoyeurs est la redondance avec MIC comme codage primaire et DVI comme codage secondaire. Les codages ne peuvent pas être spécifiés dans l'attribut fmtp si ils ne sont pas aussi spécifiés dans des codages valides sur la ligne de support ("m=").

6 Considérations pour la sécurité

Les paquets RTP qui contiennent des informations redondantes sont sujets aux considérations sur la sécurité exposées dans la spécification RTP [2], et dans tout profil RTP approprié (par exemple [3]). Cela implique que la confidentialité des flux de support est réalisée par le chiffrement. Le chiffrement d'un flux de données redondant peut survenir de deux façons :

1. Le flux entier est à sécuriser, et tout participant est supposé avoir les clés pour décoder le flux entier. Dans ce cas, rien de particulier n'est à faire, et le chiffrement est effectué de la façon habituelle.
2. Une portion du flux est à chiffrer avec une clé différente du reste. Dans ce cas, une copie redondante du dernier paquet de cette portion ne peut pas être envoyée, car il n'y a pas de paquet suivant chiffré avec la clé correcte dans lequel l'envoyer. Des limitations similaires peuvent survenir lors de l'activation/désactivation du chiffrement.

Le choix entre les deux ne concerne que le codeur. Les décodeurs peuvent déchiffrer l'une ou l'autre forme sans modification.

Bien que l'ajout de la redondance à faible bande passante à un flux audio soit un moyen efficace de protéger ce flux contre la perte de paquet, les concepteurs d'application devraient être conscients que l'ajout de grandes quantités de redondance va augmenter l'encombrement du réseau, et donc la perte de paquets, ce qui empire le problème qu'est censée résoudre l'utilisation de la redondance. Au pire, cela peut conduire à un encombrement excessif du réseau et peut constituer une attaque de déni de service.

7 Exemple de paquet

Un paquet de données audio RTP qui contient un primaire DVI4 (à 8 kHz) et un seul bloc de redondance codé en utilisant un CPL à 8 kHz (tous deux avec des paquets à 20 ms) comme défini dans le profil audio/vidéo RTP [3] est illustré comme suit :

```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|V=2|P|X| CC=0 |M|          TCU          | Numéro de séquence du primaire|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|          Horodatage du codage primaire          |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|          Identifiant de la source de synchronisation (SSRC)          |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|1|TCU du bloc=7| Décalage d'horodatage          | Longueur du bloc |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|0|TCU du bloc=5|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|
+          Données redondantes codées en CPL (PT=7)          +
|          (14 octets)          |
+          +-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|
+          Données primaires codées en DVI4 (PT=5)          +
+          (84 octets, pas à l'échelle)          +
/
+          +-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

8 Adresse des auteurs

Colin Perkins/Isidor Kouvelas/Orion Hodson/Vicky Hardman
Department of Computer Science
University College London
London WC1E 6BT
United Kingdom
mél : {c.perkins|i.kouvelas|o.hodson|v.hardman}@cs.ucl.ac.uk

Mark Handley
USC Information Sciences Institute
c/o MIT Laboratory for Computer Science
545 Technology Square
Cambridge, MA 02139, USA
mél : mjh@isi.edu

Jean-Chrysostome Bolot/Andres Vega-Garcia/Sacha Fosse-Parisis
INRIA Sophia Antipolis
2004 Route des Lucioles, BP 93
06902 Sophia Antipolis
France
mél : {bolot|avega|sfosse}@sophia.inria.fr

9 Références

- [1] V.J. Hardman, M.A. Sasse, M. Handley and A. Watson "Reliable Audio for Use over the Internet", Proceedings INET'95, Honolulu, Oahu, Hawaii, septembre 1995. <http://www.isoc.org/in95sprc>
- [2] H. Schulzrinne, S. Casner, R. Frederick et V. Jacobson, "RTP : protocole de transport pour applications en temps réel", RFC1889, janvier 1996. (*Obsolète, voir RFC3550 STD64*)
- [3] H. Schulzrinne, "Profil RTP pour conférences audio et vidéo avec contrôle minimal", RFC1890, janvier 1996. (*Obsolète, voir RFC3551*) (P.S.)
- [4] M. Yajnik, J. Kurose and D. Towsley "Packet loss correlation in the Mbone multicast network", IEEE Globecom Internet workshop, London, novembre 1996
- [5] J.-C. Bolot and A. Vega-Garcia; The case for FEC-based error control for packet audio in the Internet; ACM Multimedia Systems, 1997
- [6] M. Handley et V. Jacobson, "SDP : Protocole de description de session", RFC2327, avril 1998. (*Obsolète; voir RFC4566*)
- [7] S. Bradner, "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", RFC2119, BCP 14, mars 1997.