

# RFC 5795 : The RObust Header Compression (ROHC) Framework

Stéphane Bortzmeyer  
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 4 avril 2010

Date de publication du RFC : Mars 2010

<https://www.bortzmeyer.org/5795.html>

---

Quels que soient les progrès des technologies, il n'y a jamais assez de **capacité**. Même aujourd'hui, les vieux modems restent en service (y compris dans un pays riche, mais étendu, comme les États-Unis) et certaines technologies récentes offrent une capacité limitée (par exemple sur les téléphones mobiles). Il est donc utile de pouvoir comprimer les données et aussi les **en-têtes** des paquets émis, qui sont souvent très redondants, ouvrant ainsi de bonnes perspectives pour la compression. Plusieurs mécanismes de compression ont été inventés depuis les débuts de l'Internet et le projet ROHC ("*Robust Header Compression*") a été créé pour mettre de l'ordre dans ces mécanismes, en développant un cadre commun. Ce RFC spécifie ce cadre.

La section 1 du RFC, l'introduction, donne quelques chiffres sur les gains qu'on peut tirer de la compression. Par exemple, si on utilise RTP (RFC 3550<sup>1</sup>) pour transporter de la voix sur IP, il faudra transporter les 40 octets de l'en-tête (en IPv6), les 12 octets de l'en-tête UDP et les 8 octets de l'en-tête RTP (sans compter les en-têtes de la couche liaison). Ainsi, un paquet de données de 20 octets (une taille courante en voix sur IP) pourrait nécessiter 60 octets d'en-têtes TCP/IP.

Le RFC originel sur ROHC était le RFC 3095 qui incluait à la fois le cadre général et les **profils** de compression pour certains protocoles (un profil est un protocole de compression donné, adapté à un certain protocole). Une réforme générale de ROHC a eu lieu et il est désormais normalisé dans plusieurs RFC, notre RFC 5795 pour le cadre général, les RFC 4996 et RFC 5225 pour des profils pour, respectivement, TCP et les protocoles sans connexion comme IP ou UDP, le RFC 4997 pour le langage formel de définition des compressions, etc. (Le RFC 3095 n'est pas officiellement abandonné puisque le protocole est le même, il est juste mieux décrit.) Le cadre général de ROHC était initialement dans le RFC 4995, mais il comportait quelques bogues, corrigées par notre RFC qui remplace donc le RFC 4995.

---

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc3550.txt>

La section 3 du RFC décrit les bases de la compression. Le principe est que le **compresseur** ne transmette pas certaines informations, car elles n'ont pas changé, ou bien peuvent être déduites automatiquement. Cela implique que le **décompresseur** puisse se souvenir de l'état de la compression, ce que ROHC nomme le **contexte** (la section 2.2 décrit toute la terminologie). Le maintien de ce contexte, même en présence de perturbations sur le réseau, est le principal défi de la compression. Sur des liens de mauvaise qualité (par exemple avec beaucoup de pertes), les mécanismes de compression pré-ROHC se comportaient plutôt mal (voir aussi la section 1).

Naturellement, rien n'est gratuit : le gain en capacité du réseau sera obtenu au détriment de la puissance de calcul, puisque compression et décompression nécessitent des calculs. La vitesse des processeurs des téléphones portables augmentant plus vite que leur capacité réseau, le choix de la compression semble raisonnable.

La section 3.2 rappelle l'histoire de la compression. Le premier grand travail dans le monde TCP/IP avait été la compression des en-têtes TCP par Van Jacobson en 1990 (RFC 1144), dont tous les utilisateurs de SLIP dans les années 1980 et 90 se souviennent (« Tu es sûr d'avoir activé la compression VJ? »).

La section 4 décrit de manière relativement informelle le fonctionnement de ROHC, notamment (section 4.1), les différentes classes de changement dans les en-têtes, qui permettent leur prédiction par le décompresseur. Ainsi, la classe `STATIC` décrit les informations qui ne changent pas pendant la durée du flux de données (le numéro de version IP par exemple), la classe `INFERRED` étant composée des en-têtes qui peuvent se déduire ou se calculer à partir d'autres informations (comme la taille du paquet).

La vraie définition de ROHC est en section 5. C'est là qu'on trouve, par exemple, la description du mécanisme de rétroaction (section 5.2.4) qui permet au décompresseur de tenir le compresseur au courant du succès (ou de l'échec) de son travail. Autre exemple (section 5.1.2), chaque canal de communication ROHC a, parmi ses paramètres, un identificateur du profil utilisé (la liste des identificateurs possibles est dans un registre IANA <<https://www.iana.org/assignments/rohc-pro-ids/rohc-pro-ids.xhtml>>, voir également la section 8). La section 5.4 normalise ainsi le profil d'identificateur `0x0000`, le profil qui ne fait rien, les profils « actifs » étant définis dans d'autres RFC.

Les changements par rapport au RFC 4995 sont minimes mais peuvent affecter l'interopérabilité, puisqu'il y a correction d'une bogue : l'encodage de la rétroaction (section 5.2.4.1) a été refait (le RFC 4995 avait introduit une différence par rapport au RFC 3095) et une nouvelle section 5.4.3 a été faite pour décrire l'initialisation du contexte dans le cas où on utilise le profile `0x0000` (profil sans compression).

Des déploiements de ROHC sont déjà faits dans les téléphones portables. Il existe une implémentation libre, RObust Header Compression (ROHC) library <<https://launchpad.net/rohc/>> (merci à Didier Barvaux pour cela), tirée d'une plus ancienne bibliothèque <<http://rohc.sourceforge.net/>>. La section 6 du RFC 3095 contient une excellente section sur les problèmes concrets de mise en œuvre de ROHC (section qui ne semble pas avoir été reprise dans les RFC plus récents).