

# RFC 3704 : Ingress Filtering for Multihomed Networks

Stéphane Bortzmeyer

<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 24 mars 2006

Date de publication du RFC : Mars 2004

<https://www.bortzmeyer.org/3704.html>

---

L'usurpation d'adresse IP est une des plaies de l'Internet et est à la base de nombreuses attaques. Ce RFC décrit les mécanismes de filtrage qui peuvent limiter les risques d'usurpation, dans le cas particulier de réseaux ayant plusieurs connexions à Internet ("*multihoming*").

Avant notre RFC, le RFC 2827<sup>1</sup> s'était déjà attaqué au problème et proposait de systématiser le filtrage des adresses IP source par les FAI et les opérateurs réseau.

Le RFC 2827 notait que le filtrage simple (interdire les paquets dont l'adresse IP source n'est pas attribuée à ce réseau par le FAI) ne marchait pas en présence d'un réseau ayant plusieurs connexions à Internet ("*multihomed*"). Dans de tels réseaux, il est parfaitement possible qu'un paquet ayant une adresse IP source du fournisseur d'accès A veuille sortir par le réseau du fournisseur d'accès B. Et il ne devrait pas être filtré dans ce cas.

Notre RFC (également connu sous son numéro de bonne pratique, BCP84) s'attaque donc au cas de ces réseaux. Il propose cinq méthodes :

- ACL en entrée du réseau du FAI ("*Ingress Access Lists*"). C'est la méthode du RFC 2827 et elle ne marche pas si on est "*multihomé*".
- Contrôle strict du routage inverse ("*Strict Reverse Path Forwarding*" ou Strict RPF). Cela consiste simplement à n'accepter un paquet que si un paquet destiné à l'adresse indiquée comme source aurait été envoyé sur la même interface. Si le routage est symétrique, cela fonctionne.
- Contrôle d'un chemin possible pour le routage inverse ("*Feasible Path Reverse Path Forwarding*"). Cette technique prend en compte le fait que la plupart des protocoles de routage peuvent gérer plusieurs chemins vers une même destination et qu'il faut donc considérer tous ces chemins, pas juste la route choisie.

---

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc2827.txt>

- Contrôle approximatif du routage inverse ("*Loose Reverse Path Forwarding*"). Cette technique se contente de vérifier qu'une route existe, sans qu'elle passe forcément par l'interface par laquelle le paquet est entré.
- Contrôle approximatif sans route par défaut ("*Loose Reverse Path Forwarding ignoring default routes*"). Identique, à la précédente, cette dernière ignore en outre la route par défaut.

Notre RFC décrit ensuite en détail les forces et les faiblesses de chacune de ces cinq solutions, les cas où elles s'appliquent le mieux et le meilleur endroit du réseau où les appliquer.

Si le routeur est une machine Linux, le "*Strict RPF*" (celui qui ne convient pas en cas de routage asymétrique) peut se mettre en œuvre ainsi :

```
echo 1 > /proc/sys/net/ipv4/conf/rp_filter
```

(Sur une vieille Debian, c'est la variable `spoofprotect` dans `/etc/network/options` qui indique aux scripts de démarrage d'effectuer la commande ci-dessus. Sur une Debian récente, ce fichier a disparu et on édite désormais `/etc/sysctl.conf` pour changer la variable `net.ipv4.conf.default.rp_filter`.) (Une bonne description plus complète de configuration d'une machine Linux "*multihomé*" est <<http://wiki.wlug.org.nz/SourceBasedRouting>>.)

Sur une machine FreeBSD, c'est avec `ipfw` qu'on peut faire ce "*Strict RPF*" avec l'option `verrevpath` (et le "*Loose RPF ignoring default route*" avec l'option `versrcreach`) :

```
ipfw add 100 pass log all from any to any via ${extif} verrevpath
```