

# RFC 7048 : Neighbor Unreachability Detection Is Too Impatient

Stéphane Bortzmeyer  
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 8 janvier 2014

Date de publication du RFC : Janvier 2014

<https://www.bortzmeyer.org/7048.html>

---

Le service de découverte des voisins d'IPv6, normalisé dans le RFC 4861<sup>1</sup>, inclut une fonction de détection de l'injoignabilité du voisin (section 7.3 du RFC 4861). En trois secondes, elle permet de détecter la panne d'un voisin et donc, s'il existe un autre voisin pouvant assurer la même fonction, de basculer vers un voisin qui marche. Seulement, s'il n'existe pas de voisin équivalent, il ne sert à rien de détecter la panne (puisque l'on n'a pas d'alternative) et ce service est, dans ce cas, bien trop impatient, menant à une charge inutile du réseau.

Tout mécanisme de détection de la panne d'un composant réseau fait face à la même nécessité de compromis : si on essaie très souvent, on détectera les pannes rapidement mais on chargera le réseau. Autre problème, on risque de croire à tort qu'il y a une panne alors qu'on avait un problème temporaire dans les couches basses (convergence "*spanning tree*" prenant plusieurs secondes, par exemple). Si on essaie rarement, on épargnera le réseau, on risquera moins de détecter de fausses pannes, mais on risque de ne pas voir les vraies pannes aussi vite.

Le RFC 4861 optimise plutôt du côté de la rapidité de la détection. Un cas typique où on peut avoir un voisin équivalent est celui de deux routeurs sur le même réseau local. Il est important de détecter la panne d'un routeur très vite, pour pouvoir passer à l'autre avant que les applications ne s'en aperçoivent. Les délais spécifiés dans la section 10 du RFC 4861 (`MAX_UNICAST_SOLICIT`, trois transmissions séparées par une seconde) sont donc raisonnables. Mais ce nouveau RFC se focalise sur le cas où il n'y a pas d'alternative et où une détection agressive et répétée de la joignabilité est donc inutile. Il faut donc faire un compromis différent. Le RFC 6583 décrit ce problème, ainsi que d'autres questions opérationnelles liées à la découverte des voisins.

---

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc4861.txt>

Notez que le protocole équivalent pour IPv4, ARP (RFC 826), n'a pas ce problème car le RFC ne spécifie pas de délais particuliers, les mises en œuvre d'ARP sont donc libres d'optimiser selon leur goût, par exemple vers moins de nervosité dans la détection des pannes.

La section 3 décrit les changements faits dans le protocole. On a désormais le droit d'envoyer plus de `MAX_UNICAST_SOLICIT` s'il n'existe pas de voisin alternatif. Dans ce cas, on est censé utiliser un repli exponentiel entre deux transmissions. Le modèle de la section 7.3.2 du RFC 4861 est mis à jour pour ajouter un état : *"UNREACHABLE"*. On y entre lorsqu'il n'y a plus de réponses aux sollicitations. Lorsqu'un voisin est dans cet état, on continue à lui envoyer des paquets, comme dans l'état *"PROBE"*, mais les tests de joignabilité utilisent le *"multicast"*, car le voisin a peut-être changé d'adresse MAC. Et, évidemment, on ne choisit plus ce voisin comme routeur par défaut.

Au bout d'un moment (le RFC recommande un maximum de soixante secondes), le voisin est considéré comme définitivement mort et retiré de la table des voisins. Au fait, pour voir cette table, sur Linux, c'est (ici avec trois routeurs possibles) :

```
% ip -6 neighbour show
fe80::641 dev eth0 lladdr 00:07:b4:02:b2:01 router DELAY
fe80::250:3eff:fe97:7c00 dev eth0 lladdr 00:50:3e:97:7c:00 router STALE
fe80::20e:39ff:fe43:6400 dev eth0 lladdr 00:0e:39:43:64:00 router REACHABLE
```

Sur une autre machine, avec plusieurs voisins, dont un routeur :

```
% ip -6 neighbour show
fe80::21e:8cff:fe7f:48fa dev eth0 lladdr 00:1e:8c:7f:48:fa REACHABLE
fe80::a2f3:c1ff:fec4:5b6e dev eth0 lladdr a0:f3:c1:c4:5b:6e REACHABLE
fe80::f6ca:e5ff:fe4d:1f41 dev eth0 lladdr f4:ca:e5:4d:1f:41 router REACHABLE
fe80::f6ec:38ff:fef0:d6f9 dev eth0 lladdr f4:ec:38:f0:d6:f8 REACHABLE
```

Sur ce même Linux, les paramètres de la détection d'injoignabilité sont réglables avec `sysctl`, variables `net.ipv6.neigh.default.ucast_solicit` et `net.ipv6.neigh.default.mcast_solicit`.

Si vous aimez la programmation, la section 4 de notre RFC contient un algorithme (spécifié en langage naturel) mettant en œuvre les nouvelles possibilités, et donc moins impatient que l'algorithme actuel.