

RFC 1498 : On the Naming and Binding of Network Destinations

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 2 Juillet 2009

Date de publication du RFC : Août 1993

<http://www.bortzmeyer.org/1498.html>

Les discussions d'architecture des réseaux à l'IETF ou dans d'autres communautés techniques acheminent souvent sur le vocabulaire. Qu'est-ce que le **nom** d'une machine ? Qu'identifie t-il réellement ? La clé publique SSH d'une machine est-elle un nom ? D'ailleurs, faut-il identifier les machines ? Pourquoi pas les processus ? Ou les utilisateurs ? Et l'**adresse** ? Quel est son statut ? <http://www.foo.example:8080/toto> est-il une adresse bien qu'il inclus un nom ? Il ne s'agit pas uniquement de pinaillage philosophique. C'est en partie à cause du flou sur les concepts que les discussions d'architecture tournent en rond et s'éternisent.

D'innombrables documents, dont ce RFC, ont été produits pour tenter de clarifier la situation. Mais aucune terminologie cohérente n'a été adoptée. Notre RFC 1498¹ est donc une étape, une des plus souvent citées, dans cette longue liste. Quels sont ses points essentiels ? (Attention si vous distribuez ce document, c'est un des rares RFC qui a une licence qui n'autorise pas une distribution libre.)

Le RFC essaie d'appliquer les principes des systèmes d'exploitation aux réseaux. Les systèmes d'exploitation, eux aussi, doivent identifier les fichiers, les utilisateurs, les disques, les partitions, les parties de la mémoire, etc. Le RFC reprend la terminologie classique (mais très peu appliquée en pratique) de John Shoch, que tous les étudiants des années 1980 et 1990 ont apprise par cœur (Shoch, « *A note on Inter-Network Naming, Addressing, and Routing* »), une copie est disponible sous le nom d'IEN 19 (<<http://www.postel.org/ien/pdf/ien019.pdf>>) :

- Un **nom** identifie ce qu'on veut,
- Une **adresse** identifie où cela se trouve,
- Une **route** identifie comment y aller.

1. Pour voir le RFC de numéro NNN, <http://www.ietf.org/rfc/rfcNNN.txt>, par exemple <http://www.ietf.org/rfc/rfc1498.txt>

À noter que cette terminologie, dans notre RFC, ne ne fait pas référence à la **forme** des identificateurs. Ainsi, un nom peut aussi bien être une suite de lettres ayant un sens qu'une série de chiffres opaques (contrairement à une vision trop simpliste <<http://www.bortzmeyer.org/pourquoi-le-dns.html>> selon laquelle les noms sont juste une forme conviviale des adresses). La terminologie de ce RFC ne sépare donc pas les identificateurs qu'un humain peut traiter et ceux qu'une machine peut traiter.

On peut voir tout de suite (ce point ne figure pas dans le RFC) que cette terminologie ne s'applique pas du tout à l'Internet. Une adresse IP, par exemple, n'identifie pas une position dans la topologie de l'Internet, puisqu'il existe des adresses PI, il y a l'"*anycast*", etc. Un URL est un mélange de nom (la ressource que je veux) et d'adresse (puisque'il indique où trouver la ressource, avec pas mal de détails de bas niveau comme le numéro de port).

J. Saltzer, donc, procède autrement. Il distingue quatre types d'objets qu'on peut avoir envie d'identifier :

- Les services et les utilisateurs. Les services sont, par exemple, un service de synchronisation d'horloge.
- Les nœuds du réseau (en gros, les machines).
- Les points d'attachement au réseau. Ce sont les endroits où une machine se connecte et, dans la terminologie de Shoch, ce sont les adresses. Une machine IP "*multihomé*" aura ainsi plusieurs points d'attachement au réseau.
- Les chemins (ou routes) entre points d'attachement.

Et Saltzer insiste sur le fait qu'un nom peut prendre plusieurs formes (par exemple une binaire sur le câble et une lisible <<http://www.bortzmeyer.org/representation-texte.html>> pour les documentations et les fichiers de configuration), sa nature ne dépend pas de sa forme.

Là encore, même si ce point n'est pas abordé dans le RFC, on peut s'amuser à chercher une correspondance entre ces types et les identificateurs utilisés sur l'Internet :

- Les services ou utilisateurs sont typiquement identifiés par un URI (ou parfois par une forme dégénérée, uniquement le nom de domaine, comme lorsqu'on dit `fr.wikipedia.org` au lieu de `http://fr.wikipedia.org/`),
- Les nœuds n'ont pas vraiment d'identificateurs, à part dans certains protocoles comme SSH et HIP,
- Les points d'attachement sont vaguement identifiés par les adresses IP mais des techniques comme l'"*anycast*" ou des mécanismes d'allocation comme les adresses PI brouillent sérieusement les choses (avec l'"*anycast*", l'adresse IP est plutôt un identificateur de service),
- Les chemins sont la liste des adresses IP des routeurs traversés, celle qu'affiche traceroute.

Pour notre RFC, les quatre types d'objets cités ci-dessus ont chacun une **identité**. Cette identité se maintient même si l'objet change ses **liaisons** avec les autres objets. Ainsi, un service ne devrait pas changer de nom lorsqu'il migre d'une machine à une autre, une machine (un nœud) ne devrait pas changer de nom lorsqu'il migre d'un point d'attachement à un autre, etc.

L'essentiel, donc, dit le RFC, est la **liaison** ("*binding*") qui est faite entre un type d'objets et un autre. Ainsi, il doit exister un mécanisme de découverte qui permet de faire la liaison entre un service et la machine sur laquelle il opère. Un autre mécanisme de liaison permet de passer de la machine au point d'attachement. Un dernier fait passer du point d'attachement au chemin. Chacun de ces mécanismes doit également effectuer un choix, car il existe souvent plusieurs réponses.

Dans l'Internet actuel, en l'absence de vrai identificateur de machine, il y a plutôt un mécanisme unique, le DNS, qui sert aux deux premiers rôles. BGP, lui, prend en charge le dernier. (Ils ne sont pas cités dans le RFC.)

Il existe aussi parfois un autre mécanisme préalable aux autres, celui qui permet de trouver l'identificateur du service, c'est par exemple un moteur de recherche. La question est discutée dans le RFC, qui

note qu'il n'y a pas toujours une liaison explicite, dans une table, entre l'idée humaine d'un service et son nom formel. Ainsi, changer un nom formel est très difficile (voir l'excellent article "*Cool URIs don't change*" <<http://www.w3.org/Provider/Style/URI>>).

Le RFC discute ensuite quelques cas de réseaux réels, en tenant de les mettre en correspondance avec les notions de types d'objets et de liaisons. Dans le cas d'Ethernet, l'adresse MAC est très mal nommée puisqu'elle est constante et ne dépend pas de la localisation. C'est en fait plutôt un nom et elle est utilisée comme telle par certains logiciels, par exemple pour le contrôle de la licence d'utilisation. Le cas est en fait plus complexe, par exemple parce qu'une machine a deux cartes Ethernet et peut se connecter à un réseau Ethernet en deux points. Les réseaux réels n'ont pas été conçus selon une architecture propre...

Un autre exemple est le vieux NCP où l'identificateur appelé « nom » était en fait dépendant de la position de la machine. En l'absence de mécanisme de résolution commun, la liaison entre un nom et un point d'attachement était câblée en dur dans toutes les machines du réseau. Changer une machine de place imposait donc de changer son nom... ou de mettre à jour toutes les machines du réseau, ce qui n'est pas mieux. Saltzer argumente donc que la nature d'un identificateur dépend de l'existence ou nom d'un mécanisme de résolution, et donc de l'existence d'une vraie liaison avec les identificateurs des autres types d'objet (Un autre exemple aurait pu être Fidonet.)

Donc, pour résumer ce brillant document :

- Il existe plusieurs types d'objet à identifier,
- Un mécanisme de liaison est nécessaire pour passer des identificateurs d'un type aux identificateurs d'un autre type,
- Les réseaux réels comme l'Internet ne collent jamais parfaitement aux modèles.