

RFC 7304 : A method for mitigating namespace collisions

Stéphane Bortzmeyer

<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 7 juillet 2014

Date de publication du RFC : Juillet 2014

<https://www.bortzmeyer.org/7304.html>

Il arrive parfois, lors de l'utilisation d'un nom de domaine, qu'il y ait plusieurs noms qui correspondent à l'intention exprimée par l'utilisateur humain (dans les milieux ICANN, c'est souvent appelé une **collision**, terme brutal conçu pour faire peur). Par exemple, si l'utilisateur dans une entreprise `example.com` tape « `www.foobar` » dans son navigateur Web, peut-être comptait-il aller en `http://www.foobar.example.com/` en comptant sur le fait que le logiciel complètera (ce qu'on nomme une "*search list*") mais peut-être aussi voulait-il aller en `http://www.foobar/` (le TLD `.foobar` n'existe pas aujourd'hui mais, au rythme où des TLD sont créés, cela pourrait arriver un jour). Dans ces conditions, comment faire pour désambiguer? Ce très court RFC décrit une méthode... et explique pourquoi elle est fortement déconseillée.

Ce problème n'est pas spécifique aux "*search lists*". On peut aussi l'avoir lorsque plusieurs espaces de nommage sont utilisés, par exemple en cas de « racines alternatives <`https://www.bortzmeyer.org/racines-alternatives.html`> ». Les partisans des dites racines écartent souvent le problème de « collision » entre deux racines en disant « il suffira de demander à l'utilisateur son choix ». Pourquoi est-ce une mauvaise idée (mon intention était d'écrire « une idée idiote » mais le RFC dont je rends compte est plus prudent)?

La section 2 de notre RFC décrit la méthode et ses défauts : si le nom n'a qu'une signification possible, on y va, sinon on présente à l'utilisateur une liste des possibilités « voulez-vous aller en `http://www.foobar.example.com/` ou en `http://www.foobar/?` » et on lui demande de choisir. On peut mémoriser ce choix, pour éviter de demander trop souvent à l'utilisateur. Mes lecteurs techniques voient sans doute immédiatement pourquoi cela ne peut pas marcher, mais ce RFC est conçu à des fins pédagogiques, pour tordre le cou une bonne fois pour toutes à cette fausse bonne idée, qui resurgit de temps en temps.

Quels sont les problèmes avec cette approche (si vous êtes enseignant en informatique, vous pouvez faire de cette question un sujet d'interrogation, pour voir si les étudiants ont bien compris les réseaux)?

- Il n’y a pas toujours un humain disponible devant l’écran. Il y a des serveurs fonctionnant automatiquement, et des applications qui s’exécutent en arrière-plan (comme les mises à jour périodiques).
- L’humain peut ne pas savoir, notamment si le choix découle d’une tentative de hameçonnage ou autre tentative délibérée de l’induire en erreur.
- Le délai supplémentaire pendant lequel l’utilisateur réfléchit peut conduire l’application à renoncer.
- Ces demandes perpétuelles peuvent être pénibles pour l’utilisateur (songez qu’une seule page Web peut comporter des dizaines de noms de domaine).
- Mémoriser les choix résoudrait partiellement le problème précédent, mais créerait d’autres ennuis, par exemple en cas de déplacement (si l’utilisateur se servait d’un portable et qu’il quitte les locaux de `example.com`).

Bref, c’est une **mauvaise** solution et notre RFC la déconseille. Par contre, il ne propose pas d’alternative (il n’y en a pas de sérieuse).

Pour les informaticiens, notez que, question mise en œuvre technique d’une telle solution, on aurait plusieurs possibilités : intercepter les requêtes de résolution de noms dans un “*shim*”, une mince couche logicielle entre l’application et la bibliothèque de résolution de noms (c’est ainsi que sont souvent mis en œuvre des TLD « alternatifs » comme `.bit` ou `.onion`) ou bien en remplaçant complètement la bibliothèque de résolution de noms. On pourrait encore mettre ce service dans le navigateur Web (ignorant les problèmes des applications non-Web), ou dans un relais sur le trajet.