RFC Origins of Domain Names

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 10 mai 2021

https://www.bortzmeyer.org/what-are-domain-names.html

Qu'est-ce qu'un nom de domaine ? La question semble simple puisque les noms de domaine sont aujourd'hui partout, y compris sur les affiches publicitaires. Mais cette question apparemment simple soulève bien d'autres questions, que ce texte d'Edward Lewis <https://datatracker.ietf.org/doc/draft-lewis-domain-names/> étudie. Notamment, il est très difficile de trouver la définition originale de « nom de domaine ».

Par exemple, quelles sont les relations des noms de domaine avec le DNS ? Les noms de domaine ont-ils été inventés par le DNS, ou bien est-ce que le DNS a été créé pour fournir un nouveau protocole pour faire des résolutions de noms de domaine ? La question n'est pas purement philosophique, elle s'est posée ces dernières années à l'IETF lorsqu'il a fallu travailler sur des protocoles de résolution de noms autres que le DNS, et utiliser les noms de domaine dans de nouveaux contextes. Cela a été le cas, par exemple, du .onion du RFC 7686 [1]. Un nom comme sjnrk23rmcl4ie5atmz664v7o7k5nkk4jh7mm6lor2n4hxz2tos
(qui pointe vers le blog que vous êtes en train de consulter, si vous utilisez Tor) est-il un nom de domaine, bien qu'il n'utilise pas du tout le DNS ? Le débat est d'autant plus confus que, en raison de l'immense succès du DNS, même ses opposants et ceux qui prétendent offrir une solution alternative <https://www.bortzmeyer.org/dns-p2p.html> appellent souvent n'importe quel protocole de résolution de noms « DNS ».

D'autres RFC ont eu une genèse difficile car ils nécessitaient des définitions claires de certains termes et concepts, définitions qu'on ne trouvait pas dans les textes fondateurs. C'est ainsi que les RFC 4592 et RFC 5936 ont suscité beaucoup de débats terminologiques.

En outre, le problème n'est pas purement technique ; les noms de domaine sont des enjeux financiers et politiques importants, et les débats sont rudes au sujet de leur gouvernance (notez au passage que l'auteur du texte travaille à l'ICANN, mais son embauche est récente, et son document n'est pas du tout langue de bois). Ainsi, l'ICANN décide des politiques d'enregistrement dans les gTLD et, dans une

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc7686.txt

certaine mesure, à la racine des noms de domaine, mais l'IETF a aussi son mécanisme d'enregistrement, spécifié dans le RFC 6761. La délimitation de leurs pouvoirs et rôles respectifs est formalisée dans le RFC 2860 et, comme tous les textes sacrés, chacun l'interprète d'une manière différente. Notamment, le terme « "*assignments of domain names for technical uses*" » (qui sont pour l'IETF) a suscité des trésors d'exégèse : qu'est-ce qu'une « utilisation technique » ?

Pour traiter cette difficile question, la démarche de ce texte <https://datatracker.ietf.org/doc/draft-lewis-domain-names/> est **historique** : plongeons-nous dans le passé, relisons les vieux RFC et voyons ce qu'ils nous racontent. Je vous révèle la vérité tout de suite : les RFC parlaient de noms de domaine **avant** que le DNS ne soit inventé. Mais, en même temps, le concept de nom de domaine a évolué. Le texte ne propose pas de solution, ou de « définition définitive », il décrit plutôt qu'il ne spécifie. À l'origine, il avait été écrit comme "*Internet-Draft*", pour être publié sous forme de RFC mais ce projet n'a pas abouti.

Aujourd'hui, la norme de base du DNS est constituée des RFC 1034 et RFC 1035. Ces documents anciens, jamais mis à jour, sont complétés par un grand nombre de RFC plus récents, formant un cauchemar pour le programmeur <https://blog.apnic.net/2018/03/29/the-dns-camel/> qui veut écrire un serveur ou un client DNS. Et aucun des deux RFC de référence ne définit ce qu'est un **nom de domaine**...Ils citent des textes précédents (IEN-116 <https://www.rfc-editor.org/ien/ien116.txt>, RFC 799, RFC 819, et RFC 830), sans donner de définition précise, même si les concepts essentiels (la nature hiérarchique des noms, le point comme séparateur des composants dans la représentation textuelle) sont mentionnés. Mais il est clair que le concept de nom de domaine existait avant le DNS : il était utilisé par SMTP. Le RFC 788 y fait déjà allusion, des années avant la création du DNS, même si c'est seulement avec le RFC 821 que SMTP utilisera officiellement ces noms, « "*Domains are a recently [en 1982] introduced concept in the ARPA Internet mail system*" ». Et avant le DNS, il y avait déjà des protocoles de nommage utilisant ces noms de domaines, par exemple dans les RFC 819 ou surtout le RFC 830. Donc, la question est tranchée : ce n'est pas le DNS qui définit les noms de domaine. Cela a pour conséquence, entre autre, qu'il est absurde de parler d'un « DNS sur la "*blockchain*" » ou d'un « DNS pair-à-pair <https://www.bortzmeyer.org/dns-p2p.html> », sous prétexte que tel ou tel protocole de nommage utilise des noms de domaine.

Bon, mais si ce n'est pas le DNS qui a créé le concept de nom de domaine, c'est qui ? Le courrier électronique, comme semble l'indiquer les RFC sur SMTP ? Comme indiqué ci-dessus, la première occurrence du terme apparait dans le RFC 788, en novembre 1981, mais sans définition (le concept était encore en discussion). La première définition apparait dans le RFC 799 mais c'est encore très confus, le nom de domaine n'est pas encore hiérarchique. (Traditionnellement, « domaine » est utilisé pour désigner une entité administrative unique, avec ses règles propres, et ce sens est encore utilisé dans des documents comme le RFC 5598. Les gens du routage utilisent le terme « système autonome » là où ceux du courrier se serviront plutôt de « domaine ». Les noms de domaine d'aujourd'hui viennent de là. Un exemple est donné par IEN 19 <https://www.rfc-editor.org/ien/ien19.txt>.) Puis le RFC 805, en février 1982, saute le pas et introduit pour la première fois explicitement la notion de nom de domaine hiérarchique (le premier RFC sur le DNS ne sera publié qu'un an plus tard, et le DNS ne sera pas adopté instantanément, loin de là). Le RFC 819 précisera le concept, et créera en même temps celui de TLD, avec le premier domaine de tête, `.arpa`.

Le texte de Lewis note bien que cette histoire n'est que celle transcrite dans les RFC. La réalité n'est pas exactement ce qui a été écrit, les choses sont sans doute plus compliquées. Mais les points importants ne font pas de doute : les noms de domaine existaient avant le DNS, et ils ont été créés surtout pour une application spécifique, SMTP, pas pour servir de système d'identificateurs généralisé. Par exemple, le RFC sur FTP, RFC 959, bien que publié deux ans après les premiers RFC sur le DNS, ne mentionne pas une seule fois les noms de domaine.

---

https://www.bortzmeyer.org/what-are-domain-names.html

Et le concept de résolution, central dans le DNS, d'où vient-il ? La première mention semble être dans le RFC 724, quasiment dans le sens actuel, celui de trouver une information (par exemple une adresse) à partir d'un nom. Selon les auteurs des premiers RFC, interrogés par Ed Lewis, le terme venait de la programmation (par exemple la résolution des noms en adresses lors de l'édition de liens).

Bien, maintenant qu'on a renoncé à trouver **la** source unique et originale du concept de nom de domaine, voyons les variantes. Car il existe plusieurs types de noms de domaine. Déjà, il y a ceux utilisés dans le DNS. Ce sont un sous-ensemble de tous les noms de domaine, avec des restrictions spécifiques, et des règles de correspondance particulières :

— La taille maximale d'un nom est de 255 octets (j'ai bien dit octets et pas caractères), celle d'un composant de 63 caractères.
— La correspondance est faite de manière insensible à la casse.

En revanche, la limitation aux caractères ASCII et l'interdiction des symboles comme + ou & est une légende : le DNS autorise tous les octets, comme le rappelle bien le RFC 2181 (section 11). Ces restrictions existent par contre dans d'autres variantes des noms de domaine. Notez bien que j'ai écrit que le protocole DNS autorise tous les octets. Les registres de noms de domaine, eux, peuvent imposer des restrictions supplémentaires, notamment parce que les gens qui réservent des noms de domaine le font souvent pour créer des noms de machine, présentés plus loin.

Le DNS introduit deux autres concepts importants : une séparation entre la représentation `<https://www.bortzmeyer.org/representation-texte.html>` des noms aux humains, et leur encodage dans les paquets effectivement échangés. Sur le câble, les noms de domaine ne comportent pas de point, par exemple (RFC 1035, section 3.1). Et le second concept est la notion d'autorité, lié à une zone (un sous-arbre géré par les mêmes serveurs faisant autorité).

Autre sous-ensemble, les noms de machines ("*host names*"). Ce n'est pas la même chose qu'un nom de domaine `<https://www.bortzmeyer.org/host-vs-domain.html>`. Ils sont soumis à une syntaxe plus restrictive, décrite dans le RFC 952, légèrement assouplie ensuite par le RFC 1123. Là, les caractères comme le * ou le _ sont vraiment interdits. Cette règle est souvent nommée LDH, pour "*Letters Digits Hyphen*", puisque seuls les lettres (ASCII), les chiffres (arabes) et le tiret sont admis.

À propos des lettres ASCII, il faut aussi parler des noms en Unicode, puisque la planète n'est pas peuplée que de gens qui écrivent en ASCII. Ces noms internationalisés, les IDN, sont normalisés dans le RFC 5890. Pourquoi un RFC supplémentaires puisque le DNS admet tous les octets `<https://www.bortzmeyer.org/pourquoi-idn-et-pas-un-dns-unicode.html>`, même non-ASCII, dans un nom ? Parce que le DNS ne permet pas d'indiquer l'encodage, et qu'en prime ses règles d'insensibilité à la casse ne sont pas celles, bien plus complexes, d'Unicode. Enfin, certaines vieilles applications ou bibliothèques auraient pu mal réagir à la présence des caractères non-ASCII. D'où la séparation des noms en deux : les "*U-label*", le nom normal, en Unicode, et le "*A-label*", une représentation en pur ASCII, normalisée dans le RFC 3492, qui permettait de minimiser les risques de mauvaises surprises. Ainsi, lorsque `potamochère.fr` est le "*U-label*",x `xn--potamochre-66a.fr` sera le "*A-label*".

Le système IDN est donc une chose de plus dans l'univers déjà riche des noms de domaine. Pour approfondir, vous pouvez aussi lire le RFC 6055, qui traite du cas des noms en Unicode en dehors du DNS, et le RFC 4290, qui parle de règles que peuvent suivre les registres de noms de domaines.

Et, justement, un protocole proche du DNS, le Multicast DNS du RFC 6762, utilise des noms qui ne sont pas passés en punycode (la forme `xn--...`) mais utilisent un encodage plus traditionnel d'Unicode, UTF-8, tel que décrit dans le RFC 5198. (L'annexe F du RFC 6762 explique ce choix.)

Notez que le texte considère que les adresses IP sont une forme de nom de domaine, puisqu'on peut les utiliser, par exemple, dans les URL. (En théorie, on peut aussi dans les adresses de courrier électronique mais, en pratique, cela ne marche pas.)

`https://www.bortzmeyer.org/what-are-domain-names.html`

Mettons maintenant notre "*hoodie*" et nos gants, pour ressembler au "*hacker*" dans les reportages de BFM TV, et descendons dans les profondeurs du "*darknet*". Non, je plaisante, on va simplement regarder les noms des services Tor, en `.onion`. Ce blog, par exemple, est accessible avec Tor en `sjnrk23rmcl4ie5atmz664v7` S'agit-il d'un nom de domaine ? Oui, certainement. Sa syntaxe est définie dans la documentation Tor <`https://gitweb.torproject.org/torspec.git/tree/address-spec.txt`>, et est celle d'un nom de domaine. (Notez que le TLD `.onion` a été réservé par le RFC 7686.) Mais ce n'est pas du tout un nom DNS : ces noms ne sont jamais résolus par le DNS, mais par un protocole spécifique à Tor (cf. le protocole de rendez-vous <`https://gitweb.torproject.org/torspec.git/tree/rend-spec-v3.txt`>), utilisant le fait que le nom est une clé cryptographique.

Et le fichier `/etc/hosts` ? Historiquement, il servait à stocker des noms de machine et leurs adresses IP. Il fonctionne encore aujourd'hui, et sur de nombreux systèmes d'exploitation. La syntaxe des noms qu'il contient n'est pas définie très rigoureusement. La section 6 du RFC 3493 parle de ce cas.

Bref, si on veut assurer l'interopérabilité, ce n'est pas facile. Il n'y a pas **une** définition unique des noms de domaine, mais plusieurs sous-ensembles plus ou moins bien définis. La tâche des interfaces utilisateur qui permettent d'indiquer des noms de domaine n'est pas facile ! Cela rend compliquée l'addition de nouveautés. Ainsi, les IDN n'avaient pas été normalisés sans mal. Le texte de Lewis cite un dernier exemple de cette difficulté à déterminer ce qui est acceptable : un nom d'un seul composant (comme `dk`) est-il valable ? C'est clairement un nom de domaine légal pour le DNS. Il peut avoir des adresses IP associées, comme dans l'exemple. Mais il ne marchera pas pour certaines applications <`https://www.bortzmeyer.org/domaines-d-un-seul-label.html`>, comme le courrier électronique.

Sinon, quelques autres articles sur le DNS sur ce blog :

— Pourquoi le DNS, pourquoi ne pas se contenter des adresses IP ? <`https://www.bortzmeyer.org/pourquoi-le-dns.html`> (Et, comme on dit dans les médias à sensation, « la réponse va vous surprendre ».)
— Parmi les alternatives au DNS relativement sérieuses, Namecoin <`https://www.bortzmeyer.org/namecoin.html`>.
— Les premières années du nommage dans l'Internet, vues par une participante <`https://www.bortzmeyer.org/history-naming.html`>.
— Résumé d'une série d'articles sur l'histoire du DNS <`https://www.bortzmeyer.org/bind-dns-history.html`>.
— Le RFC 226, premier RFC sur les noms de machines.
— Le RFC 608, premier à décrire la mise en ligne des noms.
— Le RFC 6761 parle de noms de domaines « spéciaux », comme `.onion`.
— Très récent, le RFC 8499 donne une définition officielle de « nom de domaine », quoique bien tardive. On note que cette définition (« un nom de domaine est une liste de composants ») ne fait pas référence au DNS, montrant bien que DNS et nom de domaine sont des concepts différents.