Dossier thématique n°14

Systèmes de nommage alternatifs (au DNS)

1 État des lieux et attentes

2 Les solutions alternatives

3 Résumé

4 Conclusion

5 Annexes

Aujourd'hui, l'Internet dépend du DNS pour presque toutes ses activités. Le DNS est le mécanisme par lequel des noms de domaine comme **whois.nic.fr** sont traduits (on dit « résolus ») en informations techniques comme l'adresse IP. Certes, sans le DNS, certains services continuent à fonctionner mais ils n'intéressent guère que les techniciens les mieux accrochés[1]. Pour M. Toutlemonde, il n'est pas exagéré de dire que sans le DNS, il n'y a pas d'Internet.

[1] On lit parfois que « Pour accéder à un site Web sans le DNS, il suffit de taper son adresse IP dans le navigateur par exemple http://192.0.2.45/ ». Il y a au moins deux raisons techniques qui font que, dans la plupart des cas, cela ne suffit pas.

Ce système DNS fournit de nombreuses propriétés essentielles à l'usage, comme l'unicité des noms (**fr.wikipedia.org** ne peut désigner qu'une seule chose) et la possibilité de prouver que les données attachées à un nom sont authentiques, grâce au système DNSSEC. Sa résilience, sa capacité à continuer de fonctionner, malgré pannes ou attaques, n'a jamais été prise en défaut globalement[2].

Or, ce DNS, si sa robustesse[3], sa décentralisation, et sa (relative) simplicité en ont fait un des piliers de l'Internet, et une des raisons de son succès, ce DNS a quand même des faiblesses. Il est arborescent, ce qui veut dire qu'il dépend, aussi bien techniquement que politiquement, de sa racine (cf. RFC 2826), dont le contenu est géré par le gouvernement états-unien[4]. Pour chaque nœud de l'espace de nommage (par exemple **.ly**), le DNS dépend d'un organisme, le registre, dont la politique d'enregistrement ou de suppression peut ne pas faire l'unanimité[5]. Et, techniquement, le protocole DNS présente des faiblesses comme la possibilité, pour un attaquant, de répondre à la place du serveur légitime, et voir ses réponses acceptées[6]. Pour les problèmes techniques, il existe des solutions (DNSSEC dans l'exemple cité) mais seront-elles suffisantes, et adoptées largement ? Ne faudrait-il pas passer à un autre système, plus pair-à-pair, moins susceptible d'attaques, qu'elles soient techniques ou juridico-politiques ?

De nombreux exemples d'attaques non techniques ont été cités ces dernières années. Il y a les saisies massives de noms de domaine (notamment dans **.com**) par les autorités états-uniennes, dans le cadre de l'opération In Our Sites[7], le blocage des noms de domaine de The Pirate Bay[8], le blocage des sites « terroristes » en France[9], l'affaire Rojadirecta[10] (encore dans **.com**), la censure en Turquie[11], ou des dizaines d'autres cas.

À noter que les risques liés au filtrage via le DNS ont fait l'objet de plusieurs études dont notamment celle du Conseil Scientifique de l'Afnic[12].

[2] Alors que de nombreuses pannes locales arrivent de temps en temps.

[3] http://www.afnic.fr/fr/l-afnic-en-bref/actualites/actualites-generales/6171/show/succes-pour-la-journee-du-conseil-scientifique-sous-le-signe-de-la-resilience-8.html

[4] Qui délègue certaines tâches à des organisations comme l'ICANN ou Verisign.

[5] L'exemple de .ly a été choisi en raison d'une action du gouvernement libyen contre un domaine : http://benmetcalfe.com/blog/2010/10/the-ly-domain-space-to-be-considered-unsafe/

[6] Ce n'est pas forcément facile pour l'attaquant : voir le RFC 5452.

[7] http://en.wikipedia.org/wiki/Operation_In_Our_Sites

[8] http://www.lepoint.fr/high-tech-internet/la-justice-francaise-interdit-the-pirate-bay-05-12-2014-1887236_47.php

[9] http://rue89.nouvelobs.com/2015/03/16/terrorisme-blocage-sites-internet-a-commence-258218

[10] http://arstechnica.com/tech-policy/2012/08/government-goes-0-2-admits-defeat-in-rojadirecta-domain-forfeit-case/

[11] http://lexpansion.lexpress.fr/high-tech/turquie-la-censure-d-internet-s-etend-a-google_1504828.html

[12] http://www.afnic.fr/fr/l-afnic-en-bref/actualites/actualites-generales/6573/show/le-conseil-scientifique-de-l-afnic-partage-sur-le-filtrage-internet-par-dns.html

1 État des lieux et attentes

Si, une fois qu'on a décidé que les inconvénients du DNS l'emportaient sur ses avantages, on veut concevoir un « meilleur » système, quelles propriétés faut-il lui donner ? C'est un point essentiel car, si beaucoup de gens trouvent le DNS peu satisfaisant, ils sont rarement d'accord sur ce qu'ils voudraient à la place. Voici une liste exhaustive des propriétés qu'on souhaiterait idéalement pour un système de nommage :

- **Identificateurs parlants.** Tout le monde préfère **www.rue89.com** à BE25 EAD6 1B1D CFE9 B9C2 0CD1 4136 4797 97D6 D246.
- **Identificateurs uniques au niveau mondial.** On n'a certainement pas envie de changer ses signets ou ses cartes de visite lorsqu'on passe de France en Corée. De même, si on fait de la publicité pour **fr.wikipedia.org**, on n'a certainement pas envie de devoir ajouter « sauf si vous êtes chez le FAI Untel, auquel cas c'est fr.wp.encyclo ou sauf si vous utilisez Namecoin, auquel cas c'est fr/wikipedia ». On veut que le même nom marche partout et à coup sûr (propriété que ne fournissent pas les moteurs de recherche).
- **Identificateurs stables.** La disparition d'un URL est une des plaies du Web. On veut évidemment qu'un identificateur, donné comme référence dans un livre ou un article scientifique, soit toujours valable dix ans après.
- **Identificateurs sûrs.** Le terme est un peu flou. Disons qu'on voudrait que les mécanismes d'avitaillement et de résolution des identificateurs ne puissent pas être subvertis trop facilement par un méchant. (Comme peut l'être le DNS avec la faille Kaminsky ou comme le sont les noms de domaine dans les pays où il n'y a pas de sécurité juridique du titulaire).
- **Identificateurs résolvables.** Dans la plupart des cas, on ne s'intéresse pas à l'identificateur pour lui-même, on veut l'utiliser pour obtenir une autre information (une adresse IP, par exemple, pour pouvoir s'y connecter). Il faut donc un mécanisme de résolution, pas juste d'avitaillement (d'enregistrement). Ce point est délicat parce que, d'une certaine façon, tout type d'identificateur est résolvable. Il suffit de tout mettre dans une DHT, par exemple (en oubliant les problèmes de sécurité, cruciaux avec les DHT). Ou, au contraire de l'approche pair-à-pair de la DHT, on peut passer par un serveur Web qui fait des recherches dans une base centrale et envoie un résultat. Donc, quand on dit « identificateur résolvable », il faudrait plutôt ajouter, « de manière raisonnable » (oui, c'est très flou mais c'est clair, par exemple, qu'un serveur Web centralisé n'est pas une solution raisonnable).
- **Identificateurs enregistrables facilement, pas cher et sans possibilité de refus arbitraire.** Idéalement, on voudrait un système d'enregistrement « pair-à-pair » c'est-à-dire où il n'existe pas d'autorité jouant un rôle particulier. L'expérience prouve en effet que ces autorités tendent toujours à abuser de leur pouvoir.

Or, et c'est le point important, on ne peut pas avoir toutes ces propriétés à la fois. Par exemple, si on veut des identificateurs parlants comme **milka.fr** pour une personne prénommée Milka, on peut, même si on est de bonne foi, perdre son nom de domaine au profit d'un tiers titulaire d'une marque identique. Ces identificateurs ne seront pas stables. Autre problème de stabilité : si un identificateur est parlant, il risque d'y avoir des pressions pour le modifier, si le mot acquiert un autre sens, ou si on change d'avis (un URL comme **http://example.org/monblog/jean-michelmarc-michu-est-un-clown** posera un problème de stabilité si vous souhaitez adoucir le ton plus tard...). Des identificateurs numériques arbitraires comme **1f8efda3-df57-4fd4-b755-8808a874dd38** ne suscitent pas de convoitises, ne risquent pas de devoir être modifiés, mais ne sont plus parlants... De même, pour avoir des noms enregistrables en pair-à-pair complet, la seule méthode réaliste semble être de les tirer au hasard dans un espace de grande taille (pour éviter tout risque de collisions), ce qui les rendra très peu parlants.

Les noms de domaine sont uniques au niveau mondial, parlants, relativement stables, mais pas assez en raison des convoitises qu'ils suscitent et de l'absence de sécurité juridique pour les titulaires. Grâce au DNS, ils sont facilement résolvables et, grâce à DNSSEC, cette résolution peut être assez sûre. Aussi bien pour l'unicité que pour la sécurisation avec DNSSEC, ils sont enregistrés via une autorité, le registre, dont le contrôle fait régulièrement l'objet de conflits.

Mais, et c'est le point important, il n'existe pas d'identificateur idéal, qui aurait toutes les propriétés souhaitables (cf. RFC 1737 pour un exemple de cahier des charges pour des identificateurs idéaux). Le verrons-nous apparaître dans le futur, grâce aux progrès de la recherche fondamentale ? Peut-être. Mais l'auteur de ce rapport est sceptique : bien que cela n'ait pas encore été démontré mathématiquement, faire un système qui ait toutes ces propriétés, c'est comme de violer le premier ou le second principe de la thermodynamique. Lorsque quelqu'un arrive avec une telle proposition, il existe une infime possibilité qu'il soit un génie qui ait découvert une nouvelle voie. Mais, le plus souvent, la proposition qui semblait si séduisante s'avère erronée.

Dans l'état actuel de l'art, il faut donc regarder avec méfiance un projet qui ne dit pas clairement quelles propriétés des identificateurs on souhaite obtenir. Si les auteurs du projet ne veulent pas lister explicitement les propriétés de leur système de nommage, c'est probablement parce qu'ils ont du mal à admettre que leur système n'est pas idéal et ne fait pas tout.

Ce problème de l'impossibilité de tout optimiser à la fois est souvent présenté sous le nom de triangle de Zooko [zooko.triangle] (par exemple dans un excellent texte de Dan Kaminsky[13]). Mais le triangle de Zooko oublie plusieurs propriétés importantes d'où la liste de propriétés développée ci-dessus.

Tous ces systèmes alternatifs font donc face aux mêmes défis [bortzmeyer.nofreelunch] : fournir à l'utilisateur des propriétés qui sont inconciliables ou difficilement conciliables. Par exemple, la sécurité du nom, la capacité à garantir que les données associées à ce nom sont authentiques, peut se réaliser en utilisant des clés cryptographiques comme noms. Mais de telles clés ne sont ni mémorisables, ni pratiques à manipuler. On ne peut pas les mettre dans une publicité sur le flanc de l'autobus ! Autre contradiction, des noms uniques (sénat.fr ne désigne que le Sénat de la République française et rien d'autre) sont facilement mis en œuvre par un registre qui note les noms déposés, et s'assure donc de leur unicité, mais on n'est alors plus en pair à pair. Ces contradictions sont souvent ignorées par les promoteurs des systèmes alternatifs. Par exemple, beaucoup oublient de dire que leur système ne garantit pas l'unicité et que www.example.com pourrait donc donner des résultats différents selon l'utilisateur (c'est le cas des « racines alternatives »[14]).

Beaucoup de ces projets, souvent nommés de manière incorrecte « DNS pair-à-pair » (la plupart n'avaient rien de commun avec le DNS), n'ont pas dépassé le stade du communiqué de presse. Parmi les rares qui ont passé les épreuves de la réalisation concrète et du déploiement dans la nature, trois semblent aujourd'hui se détacher. À noter que certains peuvent utiliser des noms de domaine, mais sans forcément passer par le DNS :

- **Namecoin**, de loin celui qui a le plus de noms enregistrés (mais la nature de ces systèmes alternatifs fait qu'il est souvent difficile d'avoir des statistiques fiables), fondé sur la technologie Bitcoin, et qui dispose d'une passerelle DNS via le TLD .bit.
- **GNS, partie du système GNUnet**, qui semble le moins répandu des trois mais peut-être le plus solide techniquement.
- **et « Onion hidden services »**, utilisé par le système de protection de la vie privée Tor, et qui utilise des noms fondés sur des clés cryptographiques dans le TLD .onion (par exemple silkroad6ownowfk.onion, domaine du fameux cyber-commerçant Silk Road).

[13] http://dankaminsky.com/2011/01/13/spelunk-tri/

[14] http://www.bortzmeyer.org/racines-alternatives.html

2 Les solutions alternatives

GNUnet

GNUnet [grothoff.gns] est un ensemble de techniques permettant de profiter de l'Internet de manière « pair à pair », sans dépendre à aucun moment d'un organisme ayant un rôle privilégié dans le système. On va surtout regarder un des composants de GNUnet, GNS[15] (ex-GADS), un système de nommage « alternatif ».

GNUnet met les cartes sur table dès le début. Il propose deux systèmes de nommage dont l'un dit ouvertement qu'il renonce à l'unicité. Le premier système de nommage utilise des clés cryptographiques, le nom étant la clé publique (dans le TLD **.zkey**). Les noms sont quasiment uniques (car tirés au sort dans un espace immense), et c'est très sûr (sans utiliser du tout de registre, le détenteur de la clé privée peut facilement prouver qu'il est le titulaire). Comme indiqué plus haut, ce n'est par contre pas pratique du tout. Ce premier système sera donc rarement visible des utilisateurs.

Le second système utilise des noms qui sont contrôlés par chaque pair sur le réseau. D'une certaine façon, avec GNS, tout le monde est un registre et enregistre les noms (dans le TLD **.gnu**) qu'il veut (typiquement, ses amis et correspondants). Ces noms locaux sont sécurisés par la cryptographie. Ces noms sont relatifs et ils ne sont donc pas uniques. Par exemple, **www.sénat.gnu** est la ressource « **www** » pour ma ressource « **sénat** ». Pour un autre utilisateur, qui connait un autre Sénat (par exemple un utilisateur en Belgique[16]) cela désignera tout à fait autre chose.

On notera que ce système de noms relatifs n'a pas été inventé par GNUnet : largement utilisé dans le réseau UUCP (déployé en beaucoup d'endroits avant l'Internet), il a été théorisé par le projet de recherche SDSI[17] (Simple Distributed Security Infrastructure). Notez que ce système n'exclut pas la possibilité d'avoir des registres (il en existe déjà au moins un[18]), simplement, il permet le choix par l'utilisateur.

GNUnet a également une passerelle DNS, permettant aux logiciels actuels de résoudre les noms GNS (clés, ou noms relatifs).

GNUnet est actuellement mis en œuvre dans un logiciel libre, distribué à tous mais il semble que la communauté d'utilisateurs soit aujourd'hui très réduite[19].

GNUnet a deux modes d'utilisation : un mode très disruptif, qui nécessite de changer des habitudes (utiliser des clés cryptographiques comme noms de domaine, ou bien utiliser des noms relatifs non uniques). L'expérience prouve qu'obtenir un changement d'habitudes des utilisateurs est très difficile, et il est donc difficile de croire que ce mode puisse se populariser. Mais on peut aussi imaginer qu'un autre mode d'utilisation se répande : un système comme GNS, qui permet du pair-à-pair complet mais, comme les noms relatifs sont trop déstabilisants pour les utilisateurs, en pratique, quelques noms émergeraient, d'organisations gérant des registres, et les noms étant créés à partir de ces registres (c'est exactement ainsi qu'UUCP avait évolué). Ainsi, si « diderot » est un nom géré par un registre à qui beaucoup font confiance, le nom **sdsi.shamir.diderot** serait un nom « unique de facto ».

[15] https://gnunet.org/gns

[16] http://www.senate.be/

[17] http://people.csail.mit.edu/rivest/sdsi10.html

[18] https://gnunet.org/fcfs/

[19] Il faut dire que le logiciel n'est pas trivial à installer et surtout à configurer.

Namecoin

Namecoin [namecoin.info] repose sur un livre des opérations, une chaîne publique de blocs, comme Bitcoin (c'est d'ailleurs à l'origine le même code, mais la chaîne est différente et on ne peut pas acheter des noms avec des bitcoins). On l'oublie souvent, mais les transactions Bitcoin incluent un programme, écrit dans un langage simple et limité, exécuté pour valider la transaction. Dans Bitcoin, ce langage est très limité, notamment pour des raisons de sécurité. Il est un peu plus riche dans Namecoin, il comporte notamment des méthodes pour enregistrer un nom. L'existence d'un nom se vérifie donc en validant toute la chaîne et en relevant la création d'un nom, pas trop ancienne (les noms sont enregistrés pour une certaine période). On a donc, sinon vaincu, du moins sérieusement endommagé le triangle de Zooko : on a des noms sympathiques (on choisit le nom qu'on veut), sûrs (tout le monde a la possibilité de vérifier l'intégrité du livre des opérations[20]) et uniques (de même qu'avec Bitcoin, tout le monde peut vérifier qu'un bitcoin n'a pas été dépensé deux fois, avec Namecoin, tout le monde peut vérifier qu'un nom n'a pas été enregistré deux fois). Ce système de « transparence absolue » où tout se fait au grand jour, est à la base de la sécurité de plusieurs systèmes de l'Internet [bortzmeyer.poil]. Il existe d'ailleurs un explorateur public du livre des opérations[21].

Namecoin n'est pas gratuit. Il faut payer, en namecoins. Cette monnaie s'obtient, comme les bitcoins, en minant, ou bien en l'achetant à quelqu'un d'autre, par exemple sur une place de marché comme Kraken[22].

L'absence de registre se paie en sécurité : comme avec Bitcoin, si on perd sa clé privée, on perd tout, et sans recours possible[23]. D'autre part, avec Namecoin, les noms ne sont réservés que pour une période donnée. Pensez à les renouveler. (Et mettez en place une supervision, par exemple avec Name Alert[24].)

Tout ceci est bien plus large que le DNS actuel. Mais les applications ayant l'habitude de parler DNS, un nouveau mécanisme de nommage alternatif n'a des chances que s'il a une passerelle avec le DNS. Nous pouvons associer à ces noms des informations intéressantes comme l'adresse de courrier ou comme des adresses IP. Le principe est d'utiliser un TLD dédié, **.bit** (non officiellement enregistré, attention, des problèmes pourront survenir). Il faut monter un serveur DNS faisant autorité pour **.bit** et/ou configurer ses résolveurs pour utiliser des serveurs de **.bit**. Il existe une convention qui partitionne Namecoin en plusieurs espaces de nommage. Pour être publié dans **.bit**, le nom doit être préfixe par **d/**. M. Michu va donc enregistrer **d/michu**. Si vous avez vous-même un résolveur DNS qui gère les **.bit**, vous pouvez vérifier que cela marche :

```
% dig AAAA michu.bit
...
;; ANSWER SECTION:
michu.bit. 86357 IN AAAA 2605:4500:2:245b::42
```

Avec un serveur Web correctement configuré, vous pouvez, si votre résolveur gère **.bit**, visiter http://michu.bit/.

À noter que le livre des opérations contient une copie de toute la base des noms. Trouver les données sur un nom est donc trivial, nul besoin de whois. L'interrogation du livre des opérations peut être faite en ligne via l'explorateur public[25] ou via n'importe quelle passerelle comme DNSchain[26]. Contrairement au système actuel des noms de domaine, qui utilise deux protocoles complètement différents pour distribuer les données, le DNS et whois, Namecoin n'a qu'un mécanisme pour tout. Comme ces données sont publiques, cela permet de publier des statistiques (ce qui n'est pas possible pour les autres systèmes) : début 2014, il y avait 15 000 enregistrements Namecoin publiés. Notez que le livre des opérations inclut aussi les valeurs passées, qui ne sont jamais effaçables.

[20] On lit parfois que « M. Michu ne va pas faire ces vérifications ! » mais ce n'est pas le but. Tout le monde peut vérifier, et cette possibilité suffit à empêcher la plupart des manipulations.

[21] http://explorer.dot-bit.org/

[22] https://kraken.com/

[23] Dans le futur, des systèmes fondés sur les signatures multiples régleront peut-être partiellement ce problème.

[24] http://namealert.mvps.eu/edit

[25] http://explorer.dot-bit.org/

[26] https://github.com/okTurtles/dnschain

Comme les applications existantes ne parlent pas Namecoin, il faut une passerelle DNS➟Namecoin. Ainsi, les applications continueront à parler DNS comme avant mais interrogeront un résolveur DNS qui relaiera vers Namecoin, utilisant le TLD. Les gens les plus confiants ou les plus inconscients utiliseront un résolveur Namecoin public[27]. Les autres feront tourner un générateur de zones local qui, à partir de leur copie du livre des opérations, va fabriquer une copie locale de **.bit**, à faire charger par un serveur DNS local. À l'heure actuelle, ces deux mécanismes (résolveur public et copie locale) sont documentés mais pas du tout intégrés dans un logiciel simple d'installation et de configuration.

Une particularité intéressante de Namecoin est que les « données sociales » (celles qu'on obtient par whois dans le monde des noms de domaine) et les données techniques sont dans la même base, et récupérées par les mêmes mécanismes. Une autre particularité est que le livre des opérations contient tous les enregistrements, présents et passés. Il n'y a donc pas besoin de service qui stocke l'historique[28]. Cela permet des recherches intéressantes comme [baker.namecoin].

Tor/Onion

Le système Tor [tor.overview] est surtout connu comme un moyen d'anonymisation[29] des connexions sortantes, celles qu'on fait vers des sites externes. Si on veut visiter **http://www.opensocietyfoundations.org/** mais qu'on vit dans un pays où cela peut attirer l'attention des autorités, Tor va vous permettre d'être plus discret, en routant votre trafic par plusieurs nœuds successifs du réseau Tor (et en chiffrant le tout). Tor peut permettre aussi de contourner la censure, en évitant de donner à son FAI le nom de domaine et l'adresse IP du serveur qu'on cherche à joindre, nom et adresse qui risquent d'être filtrés.

Dans ce mode, le plus connu, Tor protège le « client », l'utilisateur chez lui. Mais si on veut héberger un site Web que certains essaieront de faire fermer, par exemple Wikileaks ? Dans ce cas, Tor fournit un autre mécanisme, les hidden services[30-31], qui permet de dissimuler la destination. Un hidden service utilise un identifiant Tor, qui permet le routage (sécurisé, comme avant) dans le réseau Tor. Pour permettre l'utilisation par les applications traditionnelles, l'identifiant peut être mis dans le TLD (non délégué) **.onion**. On pourra donc avoir son blog en, par exemple **https://http://kgquuvig3tvxmzna.onion/** ou **http://7j3ncmar4jm2r3e7.onion/**. Il y a dans les services « oignon » aussi bien des services qui sont par ailleurs accessibles « normalement », comme le moteur de recherche DuckDuckGo (**http://3g2upl4pq6kufc4m.onion/**)[32] que des services uniquement « oignon » par exemple des sites Web faits par des gens qui risqueraient leur vie si leur identité était connue.

Comme vous le voyez, l'identifiant en **.onion** est choisi aléatoirement (c'est un condensat d'une clé cryptographique) mais il existe des logiciels pour essayer systématiquement des clés jusqu'à obtenir un nom qui ressemble à ce que l'on veut, d'où des noms plus parlants comme **sonntag6ej43fv2d.onion** pour le blog de Benjamin Sonntag[33].

Accéder à ces services « oignons » nécessite un logiciel client spécial. La technique la plus simple (et donc la plus sûre) est de télécharger le Tor Browser[34], une version de Firefox modifiée pour accéder à Tor[35].

[27] Même si on a confiance dans ce résolveur public, il est en général très imprudent de faire confiance à tout l'Internet situé entre vous et ce résolveur public, surtout avec un protocole comme UDP, qui ne garantit presque rien.

[28] Comme domaintools.com pour les données obtenues par whois ou DNSDB pour les données récupérées par le DNS.

[29] Relative, comme beaucoup de systèmes prétendument « anonyme ».

[30] https://www.torproject.org/docs/hidden-services.html.en

[31] Le terme de « service caché » est très connoté négativement - alors que ces services ne sont pas cachés, n'importe qui peut y accéder - et le problème est aggravé par la presse à sensation qui en a fait le « Dark Web ». Le projet Tor envisage donc de renommer ces services https://lists.torproject.org/pipermail/tor-dev/2015-February/008256.html en « onion space ».

[32] Et même comme Facebook https://www.facebook.com/notes/protect-the-graph/making-connections-to-facebook-more-secure/1526085754298237, pourtant pas un champion de la vie privée

[33] https://benjamin.sonntag.fr/Tor-les-onion-le-darknet-a-votre-portee

[34] https://www.torproject.org/projects/torbrowser.html.en

[35] Et pour laisser fuir moins d'informations personnelles, par exemple via les préférences de langue ou l'information sur le navigateur employé.

3 Résumé

Comparaison incomplète entre les techniques de nommage

NOM	NOMS PARLANTS	SÉCURITÉ	DÉPENDANCES VIS-À-VIS DE TIERS	UNICITÉ
DNS	Noms parlants.	Robustesse prouvée au feu. Sécurité bonne si on utilise DNSSEC.	Très chargé politiquement/ juridiquement.	Noms uniques.
DNS avec racines alternatives	Noms parlants.	Serveurs racine peu ou pas gérés, sans DNSSEC.	Très chargé politiquement/ juridiquement.	Noms locaux.
GNUnet	Noms parlants, Zkeys illisibles.	Sécurité garantie par la cryptographie (donc, attention à la clé privée).	Complètement pair-à-pair.	Noms locaux, seules les Zkeys sont uniques.
Namecoin	Noms parlants.	Sécurité garantie par la cryptographie (donc, attention à la clé privée) et la transparence) Code et protocole très audités.	Complètement pair-à-pair.	Noms uniques.
Tor/Onion	Noms illisibles (mais on peut dans certains cas choisir une partie du nom).	Sécurité garantie par la cryptographie (donc, attention à la clé privée) et le système Tor. Code et protocole très audités.	Complètement pair-à-pair.	Noms uniques.

4 Conclusion

Quelles sont les chances d'un système de nommage alternatif, face au champion en place, le couple nom de domaine + DNS ? Ici, on rentre dans une partie moins descriptive et plus spéculative. Le succès ou l'échec futur de GNUnet, Namecoin ou des oignons de Tor ne dépend évidemment que très partiellement de leurs qualités ou défauts techniques. Il dépend aussi du degré de tolérance ou d'intolérance des utilisateurs à l'égard du système actuel, par exemple de l'intensification de la censure, y compris dans les pays démocratiques. Les dernières années de l'évolution de l'Internet ont plutôt été marquées par l'ossification, la tendance à ce que tout changement devienne de plus en plus difficile, comme l'illustrent les difficultés de déploiement de technologies largement reconnues comme indispensables, telles que DNSSEC ou IPv6. Un système très novateur a t-il encore ses chances ?

5

Annexes

– Glossaire

Domain Name System (DNS)
Ce sigle désigne le protocole réseau utilisé pour trouver, à partir d'un nom de domaine, de l'information, notamment des adresses IP. Parfois, il est utilisé dans un sens plus large pour désigner l'ensemble du système des noms de domaine, y compris la syntaxe de ces noms, le mécanisme d'avitaillement des noms, etc.

Distributed Hash Table (DHT)
Une DHT est un mécanisme permettant d'accéder à des informations (la valeur) indexées par une clé (qui est créée dans un espace plat, sans arborescence), de manière complètement pair-à-pair, sans qu'aucune machine ou entité ne joue un rôle indispensable. Elles sont notamment très utilisées pour BitTorrent.

Internet Corporation for Assigned Names and Numbers (ICANN)
Organisation états-unienne désignée par le gouvernement de ce pays pour servir de régulateur à certains TLD (comme .com ou .pizza) et pour instruire les demandes de modification de la racine du DNS, demandes qui sont ensuite approuvées par le gouvernement états-unien.

Top-Level Domain (TLD)
Domaine de tête comme .FR, .ORG ou .PARIS.

The onion router (Tor)
Mécanisme faisant passer le trafic IP par plusieurs relais, les « routeurs oignon » afin de mieux protéger l'identité du client et/ou du serveur.

– Références

[baker.namecoin] Chris Baker, What I Did Over My Holiday Break: Namecoin Decentralized DNS Research, 2014.

[bortzmeyer.nofreelunch] Stéphane Bortzmeyer, Inventer un meilleur système de nommage : pas si facile, 2011.

[bortzmeyer.poil] Stéphane Bortzmeyer, Tous à poil (la sécurité par la publication), 2014.

[grothoff.gns] Christian Grothoff, A Censorship-Resistant, Privacy-Enhancing and Fully Decentralized Name System , 2014.

[namecoin.info] Namecoin Project, Namecoin, 2015.

[tor.overview] Tor Project, Tor: Overview, 2015.

[zooko.triangle] Zooko Wilcox-O'Hearn, Names: Distributed, Secure, Human-Readable: Choose Two, 2001.

– RFC

[RFC 1737] K. SollinsL. Masinter, Functional Requirements for Uniform Resource Names, 1994.

[RFC 2826] , IAB Technical Comment on the Unique DNS Root, 2000.

[RFC 5452] A. HubertR. van Mook, Measures for Making DNS More Resilient against Forged Answers, 2009.

L'Afnic est le registre des noms de domaine .fr (France), .re (Île de la Réunion), .yt (Mayotte), .wf (Wallis et Futuna), .tf (Terres Australes et Antarctiques), .pm (Saint-Pierre et Miquelon).

L'Afnic se positionne également comme fournisseurs de solutions techniques et de services de registre. L'Afnic - Association Française pour le Nommage Internet en Coopération - est composée d'acteurs publics et privés : représentants des pouvoirs publics, utilisateurs et prestataires de services Internet (bureaux d'enregistrement). Elle est à but non lucratif.

Retrouvez tous les dossiers thématiques de l'Afnic :
http://www.afnic.fr/fr/ressources/publications/dossiers-thematiques/

www.afnic