Issue paper n°14

Alternative naming systems (to the DNS)

The Internet today depends on the DNS for almost all of its activities. The DNS is the mechanism by which domain names such as **whois.nic.fr** are translated (we say «resolved») into technical information such as their IP address. Although some services continue to operate without the DNS, they interest few people other than the keenest technicians[1]. For the man in the street, it is no exaggeration to say that without DNS there is no Internet.

[1] We sometimes read that «To access a website without using the DNS, just enter its IP address in the browser e.g. http://192.0.2.45/». There are at least two technical reasons why, in most cases, that is not enough.

The DNS provides many properties essential for the use of the Internet, such as the uniqueness of names (**fr.wikipedia.org** can designate only one thing) and the ability to prove that the data attached to a name are authentic, thanks to the DNSSEC system. Globally the resilience of the DNS, its ability to continue to operate despite failures or attacks, has never failed[2].

But the DNS, even if its robustness[3], decentralization, and (relative) simplicity have made it one of the pillars of the Internet, and one of the reasons for its success, still has weaknesses. It is arborescent, which means that it depends, both technically and politically, on its root (see RFC 2826), the content of which is managed by the US government[4]. For each namespace node (e.g. **.ly**), the DNS depends on an organization, the registry, whose registration or deletion policy may not be unanimously approved[5]. And technically the DNS protocol has weaknesses such as the possibility for an attacker to respond instead of the legitimate server and see its responses accepted[6]. For the technical problems, there are solutions (DNSSEC in the cited example) but will they be enough, and widely adopted? Should we not move to another system, more peer-to-peer, less susceptible to attacks, whether technical or legal-political?

Numerous examples of non-technical attacks have been cited in recent years. There have been massive seizures of domain names (including under the **.com**) by the US authorities as part of the In Our Sites[7] operation, the blocking of the Pirate Bay domain names[8], the blocking of “terrorist” sites in France[9], the Rojadirecta affair[10] (also under the **.com**), censorship in Turkey[11], and dozens of other cases.

Note that the risks associated with filtering through the DNS have been the subject of several studies, including that of the Afnic Scientific Council[12].

[2] Although many local failures occur from time to time.

[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] Which delegates certain tasks to organizations such as ICANN or Verisign.

[5] The example of .ly was chosen because of proceedings by the Libyan government against a domain: http://benmetcalfe.com/blog/2010/10/the-ly-domain-space-to-be-considered-unsafe/

[6] This is not necessarily easy for the attacker: see 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 State of play and expectations

If, once we have decided that the disadvantages of DNS outweigh its benefits, we want to design a «better» system, which properties should we give it? This is a crucial point, because while many people find the DNS unsatisfactory, they rarely agree on what they want instead. Here is a list, exhaustive, of the ideal properties for a naming system:

- **Meaningful identifiers.** Everyone prefers **www.rue89.com** rather than BE25 EAD6 1B1D CFE9 B9C2 0CD1 4136 4797 97D6 D246.
- **Identifiers that are unique worldwide.** No-one wants to change bookmarks or business cards when going from France to Korea. Similarly, if you promote **en.wikipedia.org**, you do not want to have to add «unless you are with ISP x, in which case it is en.wp.encyclo, or unless you use Namecoin, in which case it is en/wikipedia». We want the same name to work everywhere and without fail (a property that search engines do not provide).
- **Stable identifiers.** The disappearance of an URL is one of the pains of the Web. Obviously we want an identifier given as a reference in a book or a scientific article to still be valid ten years later.
- **Secure identifiers.** The term is a little vague. Let's say we would like to ensure that the identifier provisioning and resolution mechanisms cannot be too easily subverted by an ill-doer. (As the DNS can be with the Kaminsky flaw or as domain names are in countries where there is no legal certainty of the holder).
- **Resolvable identifiers.** In most cases, we are not interested in the identifier itself, we want to use it to get other information (an IP address, for example, in order to be able to connect to it). We therefore need a resolution mechanism, not just a provisioning system (of registrations). This point is tricky because, in some ways, any type of identifier is resolvable. All you need to do is put everything in a DHT, for example (forgetting the security issues, which are crucial with DHTs). Or, in contrast to the peer-to-peer approach of the DHT, you can go through a Web server that searches in a central database and sends a result. So when we say «resolvable identifier,» it would be better to add «reasonably» beforehand (which is admittedly just as vague, but it is clear, for example, that a centralized Web server is not a reasonable solution).
- **Identifiers that can be easily registered, at low cost and without any possibility of arbitrary refusal.** Ideally, the registration system would be «peer to peer», i.e. there would be no authority playing a specific role. Experience has shown that authorities always tend to abuse their powers.

However, and this is the important point, we cannot have all these properties at once. For example, if you want meaningful identifiers such as **milka.fr** for a person named Milka, even if the person is acting in good faith, they can lose the domain name in favor of a third-party holder of an identical brand. Identifiers such as these will not be stable. Other stability problem: if an identifier is meaningful, there may be pressure to change it, if the word acquires a different meaning, or if you change your mind (an URL such as **http://example.org/myblog/jean-michel-michu-is-a-clown** will pose a problem of stability if you want to tone it down later...) Arbitrary numerical identifiers such as **1f8efda3-df57-4fd4-b755-8808a874dd38** are little coveted, run little risk of needing to be modified, but are no more meaningful... Similarly, to have registerable names in full peer-to-peer, the only realistic method seems to be to draw them randomly from a large space (to avoid any risk of collision), which means they will not be meaningful at all.

Domain names are unique worldwide, are meaningful and relatively stable, but not sufficiently so because they are coveted, and entail no legal certainty for holders. Thanks to the DNS, they are easily resolvable, and thanks to DNSSEC, their resolution is relatively secure. Both for that uniqueness and the security with DNSSEC, they are registered via an authority, the registry, whose control is regularly subject to conflict.

But the fact is – and this is the important point – there is no ideal identifier with all the requisite properties (see RFC 1737 for an example of the specification for ideal identifiers). Will we see one in the future, thanks to advances in basic research? Perhaps. But the author of this report is skeptical: although it has not yet been demonstrated mathematically, making a system that has all of these properties would be akin to violating the first or second law of thermodynamics. When someone comes up with such a proposal, there is only the remotest possibility that s/he is a genius who has discovered a new way forward. Most often, the proposal that seemed so attractive turns out to be erroneous.

In the current state of the art, any project that does not clearly state the properties of the requisite identifiers must be considered with suspicion. If the proponents of the project do not want to explicitly list the properties of their naming system, it is probably because they find it difficult to admit that their system is not perfect and cannot do everything.

This problem, the fact that it is impossible to optimize everything all at the same time, is often presented under the name of the Zooko triangle [zooko.triangle] (as in an excellent text by Dan Kaminsky[13]). But the Zooko triangle omits several important properties, hence the list of properties developed above.

All of these alternative systems thus face similar challenges [bortzmeyer.nofreelunch]: providing the user with properties that either cannot be reconciled at all or only with difficulty. For example, the name's security and the ability to ensure that the data associated with it are authentic can be achieved using cryptographic keys as names. But keys such as these are neither storable nor practical to handle. You can't put them in an advertisement on the side of a bus! Another contradiction is that unique names (sénat.fr only refers to the Senate of the French Republic and nothing else) are easily implemented by a registry which records the names that have been filed, and therefore ensures their uniqueness, but this no longer a peer-to-peer system. These contradictions are often ignored by the proponents of alternative systems. For example, many of them forget to mention that their system does not guarantee uniqueness, and that www.example.com may therefore give different results according to the user (this is the case of «alternative roots»[14]).

Many of these projects, often incorrectly called «peer-to-peer DNS» (most have nothing to do with the DNS), have not gone beyond the press release stage. Among the rare ones that have passed the tests of the practical implementation and deployment in the field, three now seem to stand out. Note that some may use domain names, but without necessarily using the DNS:

- **Namecoin** is by far the one that has the most registered names (but the nature of these alternative systems often makes it difficult to obtain reliable statistics), based on the Bitcoin technology, which has a DNS gateway via the .bit TLD.
- **GNS,** which is part of the GNUnet system, seems to be the least commonplace of the three, but is perhaps the most technically sound.
- **And «Onion hidden services»** used by the Tor protection of privacy system, which uses names based on cryptographic keys in the .onion TLD (e.g. silkroad6ownowfk.onion, the domain of the infamous Silk Road cyber-crime merchant).

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

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

2 Alternative solutions

GNUnet

GNUnet [grothoff.gns] is a set of technologies allowing users to enjoy the Internet in «peer to peer» mode, without depending at any time on any organization that has a special role in the system. We are going to focus on a particular component of GNUnet, the GNS[15] (formerly GADS), an «alternative» naming system.

GNUnet puts its cards on the table right from the beginning. It proposes two naming systems, openly stating that one of which waives uniqueness. The first naming system uses cryptographic keys, the name being the public key (in the **.zkey** TLD). The names are virtually unique (since they are drawn at random from a huge namespace), and very secure (without using any registry, the holder of the private key can easily prove that s/he is the owner). As indicated above, however, it is not practical at all. This first system will therefore be rarely visible to users.

The second system uses names that are controlled by each peer on the network. In a certain way, with GNS, everyone is a registry and records the names (in the **.gnu** TLD) they want (typically, their friends and correspondents). These local names are secured by cryptography. The names are relative and therefore are not unique. For example, **www.sénat.gnu** is the «**www**» resource for my resource «**senate**». For another user who knows another Senate (for example, a user in Belgium[16]) it will designate something quite different.

Note that this system of relative names was not invented by GNUnet: widely used in the UUCP network (deployed in many places before the Internet), it has been theorized by the Simple Distributed Security Infrastructure research project (SDSI[17]). Note that this system does not exclude the possibility of having registries (there already exists at least one[18]), it simply lets the user chose.

GNUnet also has a DNS gateway, allowing existing software to resolve GNS names (keys, or relative names).

GNUnet is currently implemented in free software distributed to all, but the user community today seems to be very small[19].

GNUnet can be used in two ways: one is a highly disruptive mode because it requires changing habits (such as using cryptographic keys as domain names, or using non-unique relative names). Experience shows that getting user to change their habits is extremely difficult, so it is hard to believe that this mode will become popular. But one can also imagine another mode of use becoming widespread: a system such as GNS, which allows full peer-to-peer, but since relative names are too disconcerting for users, in practice, only a few names would emerge, with organizations managing registries, and the names being created from these registries (which is exactly how UUCP evolved). In this way, if «diderot» is a name managed by a registry trusted by many, the name **sdsi.shamir.diderot** would be a name that is «de facto unique».

[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] It must be said that the software is not easy to install and above all to set up.

Namecoin

Namecoin [namecoin.info] is based on a transaction log, a public chain of blocks such as Bitcoin (in fact it uses the same code, but the chain is different and you cannot buy names with bitcoins). It is often forgotten, but Bitcoin transactions include a program, written in a simple, limited language, executed to validate the transaction. In Bitcoin, the language is extremely limited, in particular for security reasons. It is a little richer in Namecoin, and in particular includes methods for registering names. The existence of a name is thus verified by validating the entire chain and noting the creation of a name that is not too old (names are registered for a certain period). In this way, while not defeating the Zooko triangle, the system has at least seriously damaged it: we have names that are user-friendly (you choose the name you want), safe (everyone can check the integrity of the transaction log[20]) and unique (as with Bitcoin, everyone can verify that a bitcoin has not been spent twice, with Namecoin, everyone can verify that a name has not been registered twice). This system of «absolute transparency» where everything is done openly is the basis for the security of several Internet systems [bortzmeyer.poil]. There is also a public explorer of the transaction log[21].

Namecoin is not free. You have to pay in namecoins. This currency is obtained, in the same way as bitcoins, by mining them, or by purchasing them from someone else, on a market place such as Kraken[22].

The absence of any registry is paid for in terms of security: as with Bitcoin, if you lose your private key, you lose everything, and there is no right of appeal[23]. Furthermore, with Namecoin, names are only reserved for a specific period. Remember to renew them (and set up a monitoring system, for example with Name Alert[24]).

All of this is much broader than the current DNS. But since applications that used to talking DNS, the only chance a new alternative naming mechanism has of succeeding is if it has a gateway with DNS. We can associate useful information with these names, such as a mail address or IP address. The principle is to use the special **.bit** TLD (but be careful: it has not been officially registered, and problems may occur). You have to set up a DNS server that is authoritative for **.bit** and/or configure its resolvers to use **.bit** servers. There is a convention that divides Namecoin into several namespaces. To be published in **.bit**, the name must be prefixed by **d/**. Mr Smith will therefore register **d/Smith**. If you have a DNS resolver that manages **.bit**, you can verify that it works:

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

With a properly configured Web server, if your resolver manages **.bit**, you can visit **http://smith.bit/**.

Note that the transaction log contains a copy of the entire database of names. Finding the data on a name is therefore child's play, there's no need of whois. The transaction log can be queried online via the public explorer[25] or through any gateway such as DNSchain[26]. Unlike the current system of domain names, which uses two completely different protocols to output data, the DNS and whois, Namecoin has only one mechanism for doing everything. Since the data are public, statistics can be published (which is not possible for other systems): in early 2014, users had published 15,000 Namecoin registrations. Note that the transaction log also includes past values, which can never be erased.

[20] We sometimes read that «Mr. Smith will not do the checks!» But that's not the point. Everyone can check, and that is enough to prevent most of the abuse.

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

[22] https://kraken.com/

[23] In the future, systems based on multiple signatures may partially solve this problem.

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

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

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

Since existing applications do not speak Namecoin, you need a DNS➟Namecoin gateway. In this way, applications will continue to talk DNS as before but will query a DNS resolver that will relay to Namecoin, using the TLD. The most trusting people or the most gullible will use a public Namecoin resolver[27]. The others will run a local zone generator which, based on their copy of the transaction log, will produce a local copy of **.bit**, to be loaded by a local DNS server. At present, these two mechanisms (public resolver and local copy) are documented but not at all integrated into a simple installation and configuration software system.

An interesting feature of Namecoin is that the «corporate data» (those obtained by whois in the world of domain names) and technical data are in the same database and retrieved by the same mechanisms. Another feature is that the transaction log contains all the registrations, past and present. There is therefore no need for a service that stores the history file[28]. This enables interesting searches such as [baker.namecoin].

Tor/Onion

The Tor system [tor.overview] is best known as a way of making outbound connections anonymous[29], i.e. the kind of connections you make to external sites. If you want to visit **http://www.opensocietyfoundations.org/** but live in a country where this may attract the attention of the authorities, Tor will allow you to be more discreet, by routing your traffic through several successive nodes of the Tor network (and encrypting all of the data). Tor can also be used to circumvent censorship, by avoiding having to give your ISP the domain name and IP address of the server that you are trying to access, the name and address of which may be filtered.

In this mode, the best known, Tor protects the «clients», i.e. users in their homes. But what happens if you want to host a Web site that some will try to close, such as Wikileaks? In this case, Tor provides another mechanism, the hidden services[30][31], which can be used to hide the destination. A hidden service uses a Tor identifier, which enables routing (secure, as before) in the Tor network. To allow its use by traditional applications, the identifier can be placed in the (not delegated) **.onion** TLD. You can therefore have your blog, for example, in **https://http://kgquuvig3tvxmzna.onion/** or **http://7j3ncmar4jm2r3e7.onion/**. In «onion» services, there are both services that are otherwise «normally» available, such as the search engine DuckDuckGo (**http://3g2upl4pq6kufc4m.onion/**)[32] and services that are only «onion», such as websites produced by people who would risk their lives if their identities were known.

As you can see, identifiers under the .onion TLD are chosen randomly (each one is the condensate of a cryptographic key), but software is available that can be used to systematically try keys until you obtain a name that resembles what you want, resulting in more meaningful names such as **sonntag6ej43fv2d.onion** for Benjamin Sonntag's blog[33].

Access these «onion» service requires special client software. The simplest technique (and therefore the safest) is to download the Tor Browser[34], a modified version of Firefox to access Tor[35].

[27] Even if you trust in this public resolver, it is generally very unwise to trust any Internet between you and the public resolver, especially with a protocol like UDP, which guarantees virtually nothing.

[28] Such as domaintools.com for the data obtained by whois or DNSDB for data retrieved by the DNS.

[29] Relative, like many supposedly «anonymous» systems.

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

[31] The term «hidden service» has a very negative connotation – these services are not hidden, since anyone can access them – and the problem has been compounded by the tabloid press which turned them into the «Dark Web». The Tor project therefore plans to rename these services https://lists.torproject.org/pipermail/tor-dev/2015-February/008256.html as « onion space ».

[32] Even like Facebook https://www.facebook.com/notes/protect-the-graph/making-connections-to-facebook-more-secure/1526085754298237, although the site is not noted for its defence of privacy.

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

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

[35] And to leak less personal information, e.g. via language preferences or information on the browser used.

3 Resume

Incomplete comparison of naming techniques

NAME	MEANINGFUL NAMES	SECURITY	DEPENDENCY ON THIRD PAR-TIES	UNIQUENESS
DNS	Meaningful names	Tried and tested robustness Security good if DNSSEC is used.	Politically / legally highly loaded.	Unique names
DNS with alternative roots	Meaningful names	Root servers not really managed. No DNSSEC.	Politically / legally highly loaded.	Local names
GNUnet	Meaningful names, Zkeys unintelligible	Security guaranteed by cryptography (so be careful with private keys).	Fully peer-to-peer	Local names, only the Zkeys are unique
Namecoin	Meaningful names	Security guaranteed by cryptography (so be careful with private keys) and transparency (closely audited code and protocol).	Fully peer-to-peer	Unique names
Tor/Onion	Names unintelligible (but in some cases you can choose part of the name).	Security guaranteed by cryptography (so be careful with private keys) and the Tor system Code and protocol closely audited	Fully peer-to-peer	Unique names

4 Conclusion

What are the chances of an alternative naming system compared with the current champion, the DNS + domain name pair? Here the issue is less descriptive and more speculative. The future success or failure of GNUnet, Namecoin or Tor onions obviously depends only partly on their technical qualities or defects. It also depends on the degree of tolerance or intolerance of the users towards the current system, for example with the intensification of censorship, even in democratic countries. In recent years, the development of the Internet has been marked more by fossilization, the tendency for any change to become increasingly difficult, as illustrated by the difficulties in the deployment of technologies widely recognized as being indispensable such as DNSSEC or IPv6. Does a highly innovative system still have any chance?

5

Annexes

– Glossary

Domain Name System (DNS)
Acronym for the network protocol used to find, from a domain name, information, including IP addresses. Sometimes used in a broader sense to refer to the entire domain name system, including the syntax of the names, the name provisioning mechanism, etc.

Distributed Hash Table (DHT)
A DHT is a mechanism used to access information (value) indexed by a key (which is created in a flat space without any tree structure) in a fully peer-to-peer manner, without any machine or entity playing an indispensable role. DHTs are widely used for BitTorrent in particular.

Internet Corporation for Assigned Names and Numbers (ICANN)
US organization designated by the government of that country to serve as a regulator for some TLDs (such as .com or .pizza) and to examine requests to change the root of the DNS, which are then approved by the US government.

Top-Level Domain (TLD)
Tier 1 domain-name suffixes such as .fr, .org or .paris.

The onion router (Tor)
Mechanism passing IP traffic through several relays, the «onion routers» to better protect the identity of the client and/or server.

– References

[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. « French-Only »

[bortzmeyer.poil] Stéphane Bortzmeyer, Tous à poil (la sécurité par la publication), 2014. « French-Only »

[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. Sollins, L. 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.

Afnic is the French Registry for the .fr (France), .re (Reunion Island), .yt (Mayotte), .wf (Wallis and Futuna), .tf (French Southern Territories), .pm (Saint-Pierre and Miquelon).

Afnic is also positioned as a provider of technical solutions and services for registries and registrars. Afnic (the French Network Information Centre) comprises public and private stakeholders, including government authorities, users, and Internet service providers (Registrars). It is a non-profit organisation.

Read all of our issues papers:
http://www.afnic.fr/en/resources/publications/issue-papers/

www.afnic.fr