2022

NE372 : De l’autre côté du miroir

GUIDO Yves

Esisar Grenoble-inp

15/03/2022

Table des matières

De l'autre côté du miroir

1 L'INTERNET HALL OF FAME

↳ y a-t-il des français dans le temple ?

Vint Cerf TCP-IP, ISOC, Arpanet

David Clark Architecture de l'internet, visionnaire, IAB

> ***« We reject kings, presidents and voting. We believe in rough consensus and running code »[1]***

Jon Postel IETF (RFC), ISOC, IANA, DNS, Arpanet
Principe de robustesse (general design guideline for protocol design)

> ***"Be liberal in what you accept, and conservative in what you send"[2]***

2 La DOCTRINE : L'IETF, les RFC

RFC 4677 The Tao of IETF
A Novice's Guide to the Internet Engineering Task Force
(RFC 1391 TAO original publié en 1993)[3]
Décrit :

- la hiérarchie : ISOC, IESG, IAB, IANA, RFC Editor, Secretariat
- le processus d'élaboration d'un RFC (y compris le « dress code » !)

RFC 791 Internet protocol
(1981)

Voir aussi :
RFC 2119 : Key words for use in RFCs to Indicate Requirement Levels
Mots-clés : "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL"

↳ le § 8.4.4 « Security Considerations » impose aux rédacteurs de RFC d'inclure dans leurs spécifications une section sur les vulnérabilités, les menaces et les traitement ad-hoc.[4]

---

[1] *« A Cloudy Crystal Ball- Visions of the Future »*
Proceedings of the Twenty-Fourth- Internet Engineering Task Force
Massachusetts Institute
NEARnet of Technology
Cambridge- July 13-17, 1992

[2] *RFC 791* : « …In general, an implementation must be conservative in its sending behavior, and liberal in its receiving behavior. » (1992)

[3] Voir aussi RF 6722 « Publishing the "Tao of the IETF" as a Web Page »

Important :

- *La hiérarchie : ci-dessous*
- *Ecriture de RFC : question générale de l'écriture de spécifications techniques (réseau et en général)*

  ↳ *compétences essentielles de l'ingénieur*

  Exemple : PI, RFC 2119
- *Processus d'élaboration : agile, consensuel, (moderne)*

  ↳ *applicable : académies, open-source*

  ↳ *pas toujours, pas complètement : entreprises*

- *Référence à noter pour la suite :*

  Blog de Stéphane Bortzmeyer bortzmeyer.org/

  - Synthèse de RFC (en français)
  - Spécialiste DNS

[4] *BCP72* (RFC3552 « Guidelines for Writing RFC Text on Security Considerations », Jul. 2003)

3 LA HIERARCHIE

3.1 L'ISOC Internet Society (1991)

Société « savante » et « administrative » de l'Internet

- 1er membre : Jon Postel
- Héberge l'IAB
- Supporte l'IETF
- 110 « chapitres », 80000 membres, 140 membres organisateurs

3.2 L'IAB Internet Architecture Board (1996)

Structure « opérationnelle »

- Coordonne les activités techniques (IETF)
- Gère la planification à long terme
- Conseille l'ISOC
- Régule IESG et l'IANA

3.3 L'IETF (Internet Engineering Task Force)

- Elaboration et publication de standards : les Request for comments (RFC)
- Groupe informel (sans statut), hébergé par l'ISOC
- Sujets d'études par domaine, qui commencent par des groupes de travail et des groupes de discussion informels (BoFs, ou Birds of a Feather)
- Fonctionnement : agile, selon le principe du « consensus approché »
- Dernier N° de RFC : 9187 en janvier 2022

3.4 L'IANA Internet Assigned Numbers Authority (1988, créée par Jon Postel)

En 1998, intégrée à l'ICAAN

1. Domain Names
   - Management of the **DNS Root Zone** (assignments of ccTLDs and gTLDs
   - Root Zone Management
   - Database of Top Level Domains
2. Number Resources
   - Coordination of the global IP and AS number spaces, such as allocations made to Regional Internet Registries.
   - IP Addresses & AS Numbers
   - Network abuse information
3. Protocol Assignments
   - Protocol Registries
   - Apply for an assignment
   - Time Zone Database

3.5 L'ICANN Internet Corporation for Assigned Names and Numbers (1998)

- Reprend les activités de l'IANA, en particulier la gestion des
- RIR (registres régionaux, voir chapitres IP et DNS)
- Noms de domaines de 1er niveau : TLD

- Controverse : emprise du gouvernement US

  ↳ Projets alternatifs : DOT-P2P, GNS (Gnu NS),…

Important :

Toutes ces structures sont de type « association » de droit US, souvent déclinées en « chapitres » régionaux (pays ou continent)

- *Leurs schémas de financement et de fonctionnement sont très décentralisés et entrelacés.*
- *Leurs missions sont de tous ordres :*
  - *Communication et stratégie (ISOC)*
  - *Production de standards (IETF)*
  - *Gestion opérationnelle (IAB, IANA/RIR)*

4 L'ARCHITECTURE

RFC 1958 : Architectural Principles of the Internet (1996)

Grands principes (texte 7 pages) :

- Principe « de bout en bout » (**best effort**)
  - Commutation de paquet vs circuit : +routage, mais - sécurité/confiance
- Personne ne peut l'éteindre (**disponibilité**)
- Passage à l'échelle (**scalabilité**), capacités d'évolution en général (modestie sur la prédiction du futur)
- Principe de **robustesse** (RFC 791, J. Postel)
- autres **BCP** de conception, dont un paragraphe sur la sécurité : renvoie la question aux 'extrémités')

Remarques :

- Flou sur la 'qualité de service'
- Principe de bout en bout mis à mal en pratique par les « middleboxes » (box domestiques, routeurs NAT, Firewall)

5 Travail personnel :

- Sites : ISOC, IAB, IETF, ICANN/IANA
- Biographies : V. Cerf, J. Postel, D. Clark
- RFC : 4677, 791, 2119 §8.4.4, 3552 (BCP72), 1958
- Blog Bortzmeyer : RFC 1958. Eteindre internet

Gestion des noms

1. Introduction

Objectif principal : association nom/adresse réseau

Services existants (annuaires) :

1. Locaux (fichier hosts)
2. Centralisés : NIS & NIS+, LDAP (Active Directory)
3. Distribués : Domain Name System

Pour Internet , 1 & 2 non adapté (échelle $\Rightarrow$ quantité, dynamisme)

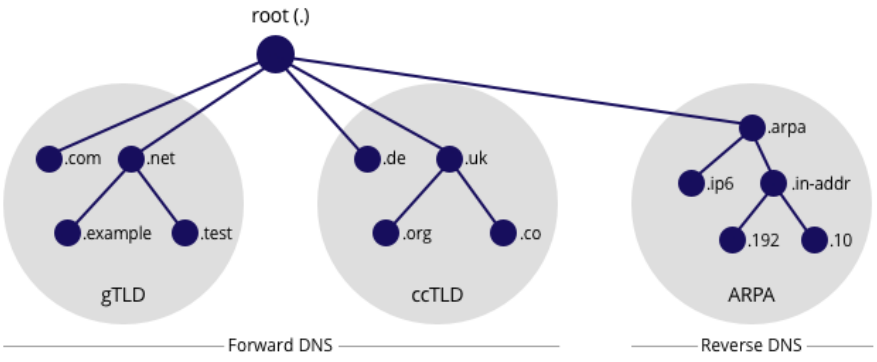
1.1 Bases

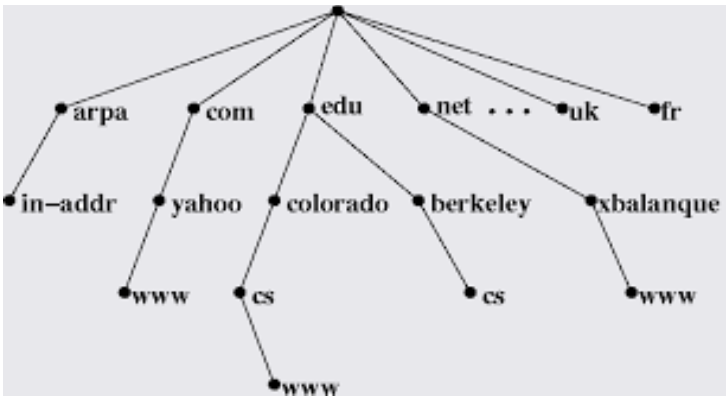
Paul Mockapetris (1984), RFC 882/883, puis RFC 1034/1035

Principes :

- Base de données distribuée (architecture et administration)
- Mécanisme client/serveur
  - **UDP (défaut) ET TCP** pour transfert de zone, et messages « longs »
- Fiabilité (réplication)
- Performances (cache, répartition de charge)

1.2 Nommage





1.3 Définitions :

- Racine : « . »
- 1er niveaux g(eneric)TLD, c(ountry)c(ode)TLD, domaine 'Arpa'
- Domaines et sous-domaines : les nœuds ou les feuilles de l'arbre
- Enregistrements : données associées à un domaine (par ex. les adresse IP)→ schéma
- Délégation : périmètre de responsabilité du nommage (sous-arbre)
- Zone : partie de l'arbre gérée par le même serveur → schéma

Fiabilité : les données (les enregistrements d'une zone) sont répliquées sur plusieurs serveurs (permet la répartition de charge). Terminologie : « *Transfert de zone* »

Performance : A chaque enregistrement est associé une durée de validité (les utilisateurs gèrent un cache)

1.4 Utilisation

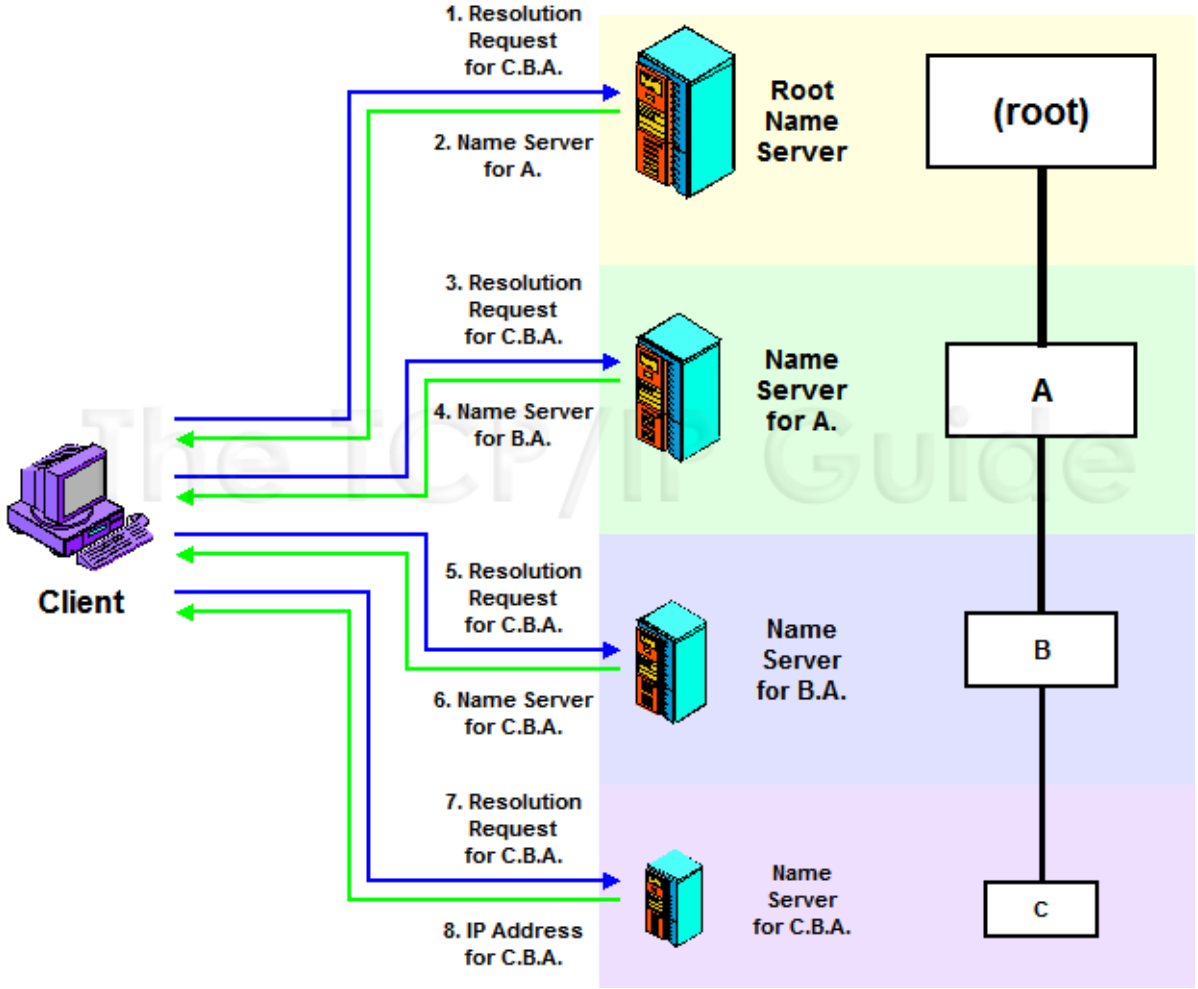
- **Quasiment toutes les applications disponibles sur l'internet utilisent un service de gestion de noms ⇒ DNS service critique !**
- A travers une API générale de résolution des noms (qui permet de gérer tous les services disponibles : hosts, nis, ldap, dns,…)

Exemple d'Unix : primitive gethostbyname(…), librairie libresolv.a et fichier resolv.conf (man resolver).

2. Fonctionnement

2.1 Fonction 'client'

Mode itératif (minimum exigé par la norme)



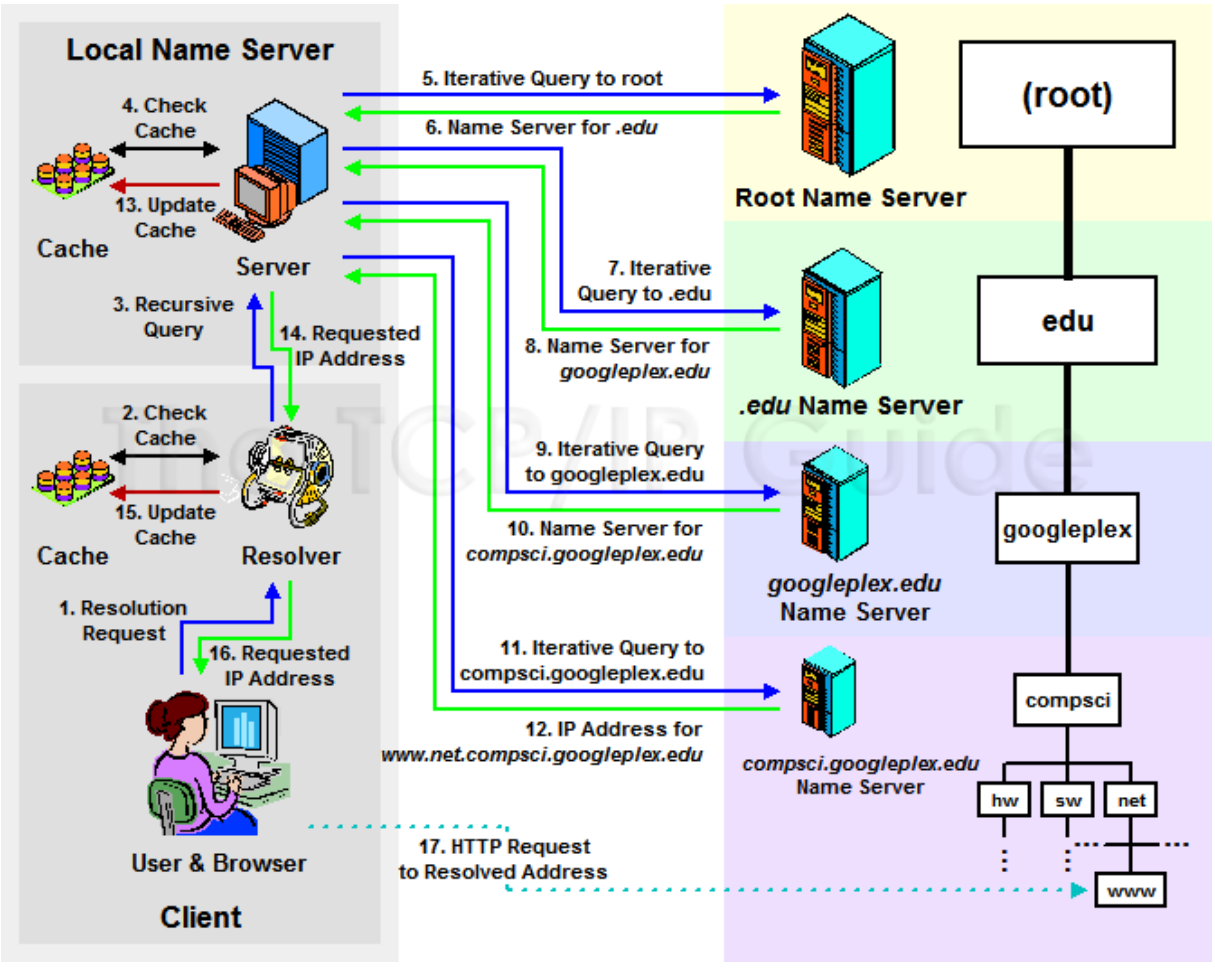
↳ Le client se débrouille avec des interrogations successives aux serveurs de zones concernés

Mode récursif

↳ Le client sous-traite sa requête au serveur qui réalise les interrogations nécessaires

↳ **Les serveurs sont eux-mêmes des clients d'autres serveurs !**

Cas d'utilisation courant : l'administrateur local ou le FAI fournissent à leurs utilisateurs un serveur récursif, qui lui-même fait de l'itératif sur les serveurs de zone concernés



↳ Récursif : coûte→ réservé aux clients 'connus' (plage d'@IP autorisées)

↳ **Il existe des serveurs récursifs publics** : Google : 8.8.8.8 et 8.8.4.4, Opendns, bornes wifi publiques ou semi-publiques (portail captifs) certains FAI,… mais aux intentions douteuses (collecte de données, tracking)

↳ **Plus « sûrs », des serveurs ouverts par la communauté :** FDN, Quad9 (utilise TLS)

↳ **Encore plus « sûr » :** installation de son propre serveur

1.1 Fonction 'serveur'

Concrètement : un 'service' sur une machine (virtuelle ou pas, mais dédiée en général)

Notion d'autorité (SOA) : comprend la zone de compétence, et la nature du serveur

primaire :

qui contient les enregistrements de référence, remplies par l'administrateur de la zone

qui fait autorité sur la zone

ou **secondaire**(s) :

qui recopie(nt) périodiquement les informations de référence du serveur primaire (à l'aide d'un n° de série)

qui sert(vent) de backup si le serveur primaire ne répond pas

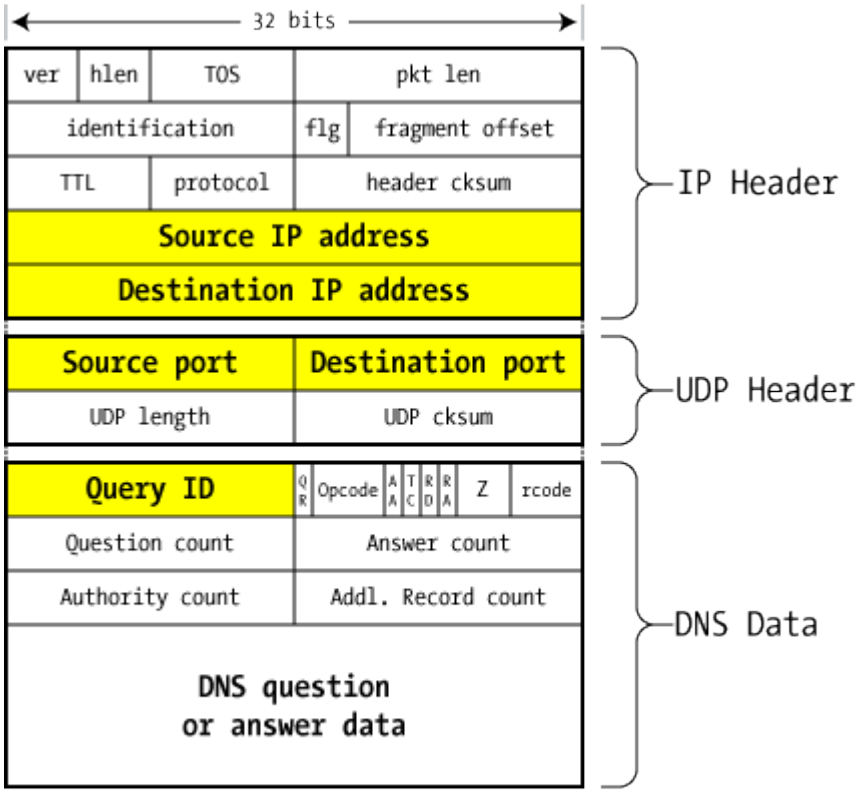
Le serveur gère un cache pour améliorer les performances :

Il ne redemande pas systématiquement les informations qu'il a obtenues récemment (sous-entendu, le serveur est lui-même client d'autres serveurs ! – voir modes itératifs ou récursifs)

1.2 Protocole de transport

UDP ou TCP : RFC 7766 (2010) oblige le support de TCP par les serveurs

En résumé **: requête par défaut en UDP**, si la réponse est trop longue[5], le serveur envoie une réponse avec un code demandant au client de passer en TCP (bit TC à 1)



DNS packet on the wire

- Historiquement, le payload est limité à 512 octets, ce qui limite notamment la taille des réponses du serveur ;
- Dans ce cas le serveur demande au client (bit TC levé à 1) de reformuler sa demande en TCP ;
- Contrainte levée (partiellement) par EDNS, une version évoluée de DNS
- UDP ou TCP ? [6]

[5] Version historique 512 octets ; EDNS : le demandeur envoie au serveur la taille max qu'il sait gérer en UDP; en pratique 4096o

[6] https://www.bortzmeyer.org/dns-over-tcp.html

2. Contenu d'une zone

Enregistrements de données :

En anglais RR=Resource Record

Forme : <nom> [TTL] [CLASSE] <TYPE> <valeur>

Exemple : ramses IN A 220.20.45.88

2.1.1 SOA record (obligatoire)

Cet enregistrement permet d'indiquer le serveur de nom maître (primaire), l'adresse e-mail d'un contact technique (avec @ remplacé par un point) et des paramètres d'expiration.

Il désigne l'autorité (**S**tart **O**f **A**uthority) ou le responsable de la zone dans la hiérarchie DNS.

Ces paramètres sont dans l'ordre : *Nom de du primaire*

wikipedia.org. IN SOA ns0.wikimedia.org. hostmaster.wikimedia.org.

2010060311 ; **n° de série**

43200 ; **Refresh** // 12h, màj secondaires

7200 ; **Retry** // 2h

1209600 ; **Expire** // 14j validité zone

3600 ; **TTL** // par défaut, 1h pour les RR

Mail de l'admin : notez le '.' A la place de l'@

Refresh : l'écart en secondes entre les demandes successives de mise à jour réalisées depuis le serveur secondaire ou les serveurs esclaves ;

Retry : le délai en secondes que doivent attendre le serveur secondaire ou les serveurs esclaves lorsque leur précédente requête a échoué ;

Expire : le délai en secondes au terme duquel la zone est considérée comme invalide si le secondaire ou les esclaves ne peuvent joindre le serveur primaire ;

Minimum ou negative TTL : utilisé pour spécifier, en secondes, la durée de vie pendant laquelle sont conservées en cache les réponses qui correspondent à des demandes d'enregistrements inexistants.

2.1.2 MX record

Une entrée DNS **MX** indique les serveurs SMTP à contacter pour envoyer un mail à un utilisateur d'un domaine donné. Par exemple :

wikimedia.org.	IN	MX	10 mchenry.wikimedia.org.
wikimedia.org.	IN	MX	50 lists.wikimedia.org.

Le serveur avec la valeur numérique la plus petite est employé en priorité. Ici, c'est donc mchenry.wikimedia.org qui doit être utilisé en premier, avec une valeur de 10.

Il n'est pas indispensable de disposer de serveurs secondaires, les serveurs émetteurs conservant les messages pendant un temps déterminé (typiquement, plusieurs jours) jusqu'à ce que le serveur primaire soit à nouveau disponible.

*Les entrées **MX** sont généralisées par les entrées **SRV** qui permettent de faire la même chose mais pour tous les services, pas seulement SMTP (le courriel). L'avantage des entrées SRV par rapport aux entrées MX est aussi qu'elles permettent de choisir un port arbitraire pour chaque service ainsi que de faire de la répartition de charge plus efficacement. L'inconvénient c'est qu'il existe encore peu de programmes clients qui gèrent les entrées SRV. Cependant, depuis 2009, avec l'augmentation de l'utilisation du protocole SIP sur les services de VoIP, les enregistrements SRV deviennent plus fréquents dans les zones DNS.*

2.1.3 A ou AAAA record

A pour les adresses IP_{v4}, **AAAA** pour les IP_{v6}

Exemples :	host1	IN	A	193.11.20.30
	www	IN	AAAA	2001:db8::3

- Plusieurs adresses IP pour un même nom (bind) :
  - Répartition de charge

    Ordre : fixed, random, cyclic (round robin)
  - Répartition géographie (GEODNS)

    Réponse en fonction de l'origine géographique de la demande : on cherche à répondre au client par l'adresse IP la plus proche de lui (au sens géographique)

2.1.4 CNAME record

L'enregistrement **CNAME** permet de créer un alias.

Par exemple :

fr.wikipedia.org. IN CNAME text.wikimedia.org.

text.wikimedia.org. IN CNAME text.esams.wikimedia.org.

text.esams.wikimedia.org. IN A 91.198.174.232

Les spécifications (RFC 1034 section 3.6.2, RFC 1912 section 2.4) recommandent de ne pas faire pointer un CNAME sur un autre CNAME ni sur un DNAME (alias pour un nom et tous ses sous-noms).

Ainsi, le premier exemple serait préférablement enregistré de la façon suivante :

fr.wikipedia.org. IN CNAME text.esams.wikimedia.org.

text.wikimedia.org. IN CNAME text.esams.wikimedia.org.

text.esams.wikimedia.org. IN A 91.198.174.232

2.1.5 TXT record

Permet de stocker des constantes de type texte

Exemple : nom IN TXT « Bonjour ! »

Utilisé pour des applications standards : DKIM par exemple, PGP pour la diffusion de clés publiques (draft) ; et pour des applications spécifiques.

2.1.6 NS record

Le record **NS** crée une délégation d'un sous-domaine vers une liste de serveurs.

Dans la zone esisar, les enregistrements **NS** suivants créent les sous-domaines irc, eis, app et p2024 esisar et délèguent ceux-ci vers les serveurs indiqués. L'ordre des serveurs est quelconque. Tous les serveurs indiqués doivent faire autorité pour le domaine : ns0 pour le domaine irc, ns1 pour eis, etc.)

irc. IN **NS** ns0.esisar.org.

eis. IN **NS** ns1.esisar.org.

app. IN **NS** ns2.esisar.org.

p2024. IN **NS** ns3.autre.tk. *(la zone déléguée "p2024" est gérée par le serveur ns3.autre.tk)*

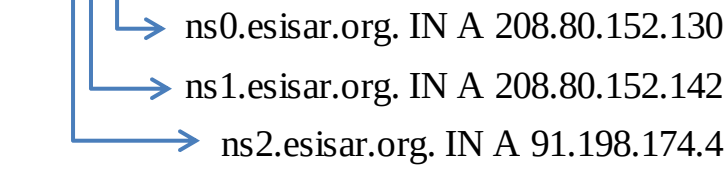
"Glue record" : quand le serveur de la zone déléguée appartient au même sous domaine (3 premiers cas ci-dessus), pour éviter les références circulaires, il est nécessaire de fournir également les adresses IP des serveurs indiqués dans la réponse (*glue records*), car ils font partie du domaine en question :

Exemple :

esisar.org. IN NS ns0.esisar.org.

esisar.org. IN NS ns1.esisar.org.

esisar.org. IN NS ns2.esisar.org.



2.1.7 PTR record « Résolution inverse »

À l'inverse d'une entrée de type A ou AAAA, une entrée **PTR** indique à quel nom d'hôte correspond une adresse IPv4 ou IPv6. Si elle est spécifiée, elle doit contenir l'enregistrement inverse d'une entrée DNS A ou AAAA.

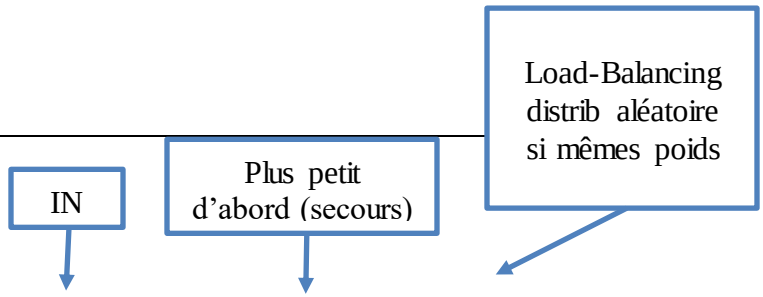
Par exemple (pour une adresse IPv4) cet enregistrement PTR est :

232.174.198.91.in-addr.arpa. IN PTR text.esams.wikimedia.org.

Correspond à cette entrée A :

text.esams.wikimedia.org. IN A 91.198.174.232

2.1.8 SRV record



Syntaxe :

```
_Service._Proto.Name TTL Class SRV Priority Weight
Port Target
```

Exemples : *_ldap._tcp.example.com* *_sip._tcp.example.com*

Service

The symbolic name of the desired service, as defined in Assigned Numbers [STD 2] or locally. An underscore (_) is prepended to the service identifier to avoid collisions with DNS labels that occur in nature.

Proto

The symbolic name of the desired protocol, with an underscore (_) prepended to prevent collisions with DNS labels that occur in nature. _TCP and _UDP are at present the most useful values for this field, though any name defined by Assigned Numbers or locally may be used (as for Service).

Name

The domain this RR refers to. The SRV RR is unique in that the name one searches for is not this name.

Priority

The priority of this target host. A client MUST attempt to contact the target host with the lowest-numbered priority it can reach; target hosts with the same priority SHOULD be tried in an order defined by the weight field. The range is 0-65535. This is a 16 bit unsigned integer in network byte order.

Weight

A server selection mechanism. The weight field specifies a relative weight for entries with the same priority. Larger weights SHOULD be given a proportionately higher probability of being selected.

Utilisations fréquentes :

- XMPP (Jabber)
- SIP
- LDAP

Exemple complet :

```
$ORIGIN example.com.
@           SOA server.example.com. root.example.com. (
              1995032001 3600 3600 604800 86400 )
            NS  server.example.com.
            NS  ns1.ip-provider.net.
            NS  ns2.ip-provider.net.
; foobar - use old-slow-box or new-fast-box if either is
; available, make three quarters of the logins go to
; new-fast-box.
_foobar._tcp    SRV 0 1 9 old-slow-box.example.com.
                SRV 0 3 9 new-fast-box.example.com.
; if neither old-slow-box or new-fast-box is up, switch to
; using the sysdmin's box and the server
                SRV 1 0 9 sysadmins-box.example.com.
                SRV 1 0 9 server.example.com.
server          A   172.30.79.10
old-slow-box    A   172.30.79.11
sysadmins-box   A   172.30.79.12
new-fast-box    A   172.30.79.13
; NO other services are supported
*._tcp          SRV  0 0 0 .
*._udp          SRV  0 0 0 .
```

2.1.9 NAPTR record

Peu répandus à l'heure actuelle (ils sont surtout utilisés par ENUM), ils décrivent une réécriture d'une clé (un nom de domaine) en URI. Par exemple, dans ENUM, des enregistrements NAPTR peuvent être utilisés pour trouver l'adresse de courrier électronique d'une personne, connaissant son numéro de téléphone (qui sert de clé à ENUM).

Ses paramètres sont dans l'ordre :

Order : indique dans quel ordre évaluer les enregistrements NAPTR ; tant qu'il reste des enregistrements d'une certaine valeur de order à examiner, les enregistrements des valeurs suivantes de order n'entrent pas en considération ;

Preference : donne une indication de priorité relative entre plusieurs enregistrements NAPTR qui ont la même valeur de order ;

Flags : indique par exemple si l'enregistrement décrit une réécriture transitoire (dont le résultat est un nom de domaine pointant sur un autre enregistrement NAPTR) ou une réécriture finale ; la sémantique précise du paramètre flags dépend de l'application DDDS ('Dynamic Delegation Discovery System', RFC 3401) employée (ENUM en est une parmi d'autres) ;

Services : décrit le service de réécriture ; par exemple dans ENUM, la valeur de services spécifie le type de l'URI résultante ; la sémantique précise de ce paramètre dépend également de l'application DDDS employée ;

Regexp : l'opération de réécriture elle-même, formalisée en une expression rationnelle ; cette expression rationnelle est à appliquer à la clé ; ne peut être fourni en même temps que replacement ;

Replacement : nom de domaine pointant sur un autre enregistrement NAPTR, permettant par exemple une réécriture transitoire par délégation ; ne peut être fourni en même temps que regexp. L'enregistrement NAPTR est défini par la RFC 3403.

2.1.10 RRSIG record

<<signatures pour DNS-SEC : hors périmètre, à compléter plus tard>>

3. Résolution inverse

Objectif : obtenir le(s) nom(s) qui correspond(ent) à une adresse

Implémentation symétrique à l'arbre de nommage : arbre de racine « in-addr-arpa » et enregistrements **PTR**.

Exemple : pour l'adresse 192.10.20.30, le 'nom' de domaine associé est : 30.20.192.in-addr-arpa.

Enregistrements :

30.20.192.in-addr-arpa IN PTR serveur.domaine.fr

31.20.192.in-addr-arpa IN PTR machine.domaine.fr

32.20.192.in-addr-arpa IN PTR machine.domaine.fr -> si la machine a plusieurs adresses

↳ Il existe des contextes administratifs plus ou moins exigeants, qui régissent la gestion de la résolution inverse d'adresses (obligations ou BCP).

↳ Attention à la sécurité (publication des adresses IP disponibles dans l'infrastructure)

4. Transfert de zone (AXFR)

Nécessaire pour répliquer les informations (enregistrements) d'un serveur de zone (ou serveur faisant « autorité », ou serveur « SOA »)

Exemples de cas de réplication d'un serveur de zone pour :

- la redondance : plusieurs serveurs qui offrent le même service ;
- la distribution géographique : serveurs identiques répartis pour performance (voir anycast))

Au niveau des standards, ce transfert doit utiliser **TCP** (alors que le mécanisme d'interrogation se fait en UDP).

Mise en œuvre :

1. Réplication périodique du fichier complet de configuration par un mécanisme externe du type FTP.

- Certains utilisent *rsync* ;
- Simple et robuste, semi-automatique ;
- Ne convient pas aux serveurs qui ont beaucoup de mises à jour (dans le temps : DynDns, ou dans la taille : ccTLD)
- Attention à la sécurité : Utiliser des mécanismes AAA, par ex. des versions sécurisées des protocoles classiques (ftps, sftp, etc.)

2. Utilisation du protocole AXFR[7][8] :

   Principe :

   - Le 'répliqué' vérifie si le numéro de série (serial number dans le RR SOA) est différent de la dernière mise à jour :
   - Dans l'affirmative, il récupère toutes les valeurs des RR du serveur de référence, par l'envoi d'une requête DNS (opcode 0) par le client avec un QTYPE (type de requête ou query type) correspondant à AXFR (valeur 252) sur une connexion TCP vers le serveur maître. Le serveur répond avec une série de messages de réponse contenant l'ensemble des ressources enregistrées (RRdata) de la zone.

3. Il existe une version incrémentale de cette synchronisation : IXFR (mais les serveurs doivent quand même savoir faire du AXFR et se replier dessus si IXFR ne fonctionne pas)

   IXFR : on ne met à jour que les RR qui ont changé :

   - peut être utile pour les serveurs de zone qui ont beaucoup de RR, par exemple pour certains serveurs ccTLD. Le serveur *.tk* gère 24.7 millions d'entrées ![9]
   - Implique l'obligation des deux parties de tenir à jour une journalisation des modifications
   - Qui implique elle-même de la complexité (et des vulnérabilités)

[7] *RFC 5936 DNS Zone Transfer Protocol*
[8] *https://*www.bortzmeyer.org/5936.html
[9] « .tk » désigne un archipel du Pacifique, Tokelau (1500 habitants). Source : https://www.verisign.com/assets/domain-name-report-Q32021.pdf

4. Planification des mises à jour :

- Elle dépend exclusivement du client (le « répliqué », en **mode « pull »**)
- Dans le cas d'un mécanisme externe comme FTP, on peut utiliser des tâches planifiées au niveau du système ;
- Le client peut aussi par exemple déclencher des transferts quand les durées de vie ont expiré ;
- Le protocole prévoit également la possibilité d'envoi par le serveur au client d'un message spécifique (NOTIFY) pour indiquer que la configuration a changée.

5. Sécurité du transfert de zone :

- Confidentialité : dans le cas d'un service « public », les informations contenues dans les RR ne sont à priori pas confidentielles. Mais l'interception des données de la zone peut révéler des informations sur l'architecture interne du serveur primaire (version des systèmes, host physique/virtuel, etc.). Ces éléments peuvent contribuer à une cartographie du système cible dans le cas d'une kill-chain.[10]

[10] *https://www.root-me.org/fr/Documentation/Reseaux/Application/Faiblesse-des-serveurs-DNS-par-transfert-de-zone*

- Intégrité : Une attaque de type MTM permet de changer les informations de la zone destinées aux serveurs secondaires, notamment les adresses IP.

- Disponibilité : Par des requêtes illégitimes, on peut tenter de saturer les serveurs (primaires et secondaires), et de neutraliser le domaine attaqué (zone autoritaire et zones déléguées !)

5. Qui fait quoi (Internet)

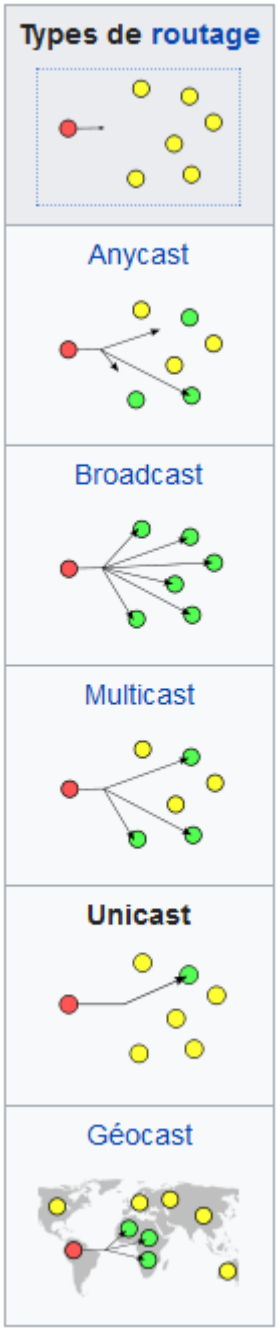
Autorité d'une zone : s'obtient de l'autorité qui gère le nœud père dans l'arbre (qui place un enregistrement NS dans son fichier de configuration)

Noms (Haut de l'arbre- TLD) : l'IANA et l'ICAAN,

- Avant 2012 : 260 ccTLD, 300 génériques ggTld
- En 2012, ICAAN autorise l'ajout de 2000 nouveaux TLD (à compter de 2014)

  $\Rightarrow$ environ **1950** TLD en février 2022 (dont 320 ccTLD)

Adresses IP : l'IANA à travers les RIR.



En 2022 l y en a **5**, dont RIPE-NCC pour l'Europe (il en existe 4 autres pour : l'Amérique, l'Asie-Pacifique, l'Amérique latine, et l'Afrique)

- Réseaux IP Européens - Network Coordination Centre
    - Le RIPE NCC a la responsabilité d'un des 13 **serveurs racine du DNS**, le **k.**root-servers.net. Il est déployé en utilisant la technologie anycast[11], c'est-à-dire que plusieurs machines physiques réparties géographiquement assurent le service, et que le routage Internet détermine celle qui traite la demande d'un client déterminé.
    - *Gestion de la zone DNS ENUM*

Le RIPE NCC gère pour la communauté mondiale la zone DNS ENUM e164.arpa. Cette branche de l'arborescence DNS globale a pour vocation d'intégrer des numéros de téléphone dans l'arborescence DNS. Ainsi, la zone 3.3.e164.arpa correspondant au préfixe téléphonique français « +33 » a été déléguée par le RIPE à l'AFNIC, qui gère déjà pour la France les noms de domaines en « .fr ».

[11] *Mis en œuvre grâce à BGP qui annonce simultanément la même tranche d'adresses IP depuis plusieurs endroits du réseau. De cette façon, les paquets sont routés vers le point le "plus proche" du réseau annonçant la route de destination.*

Opérations :

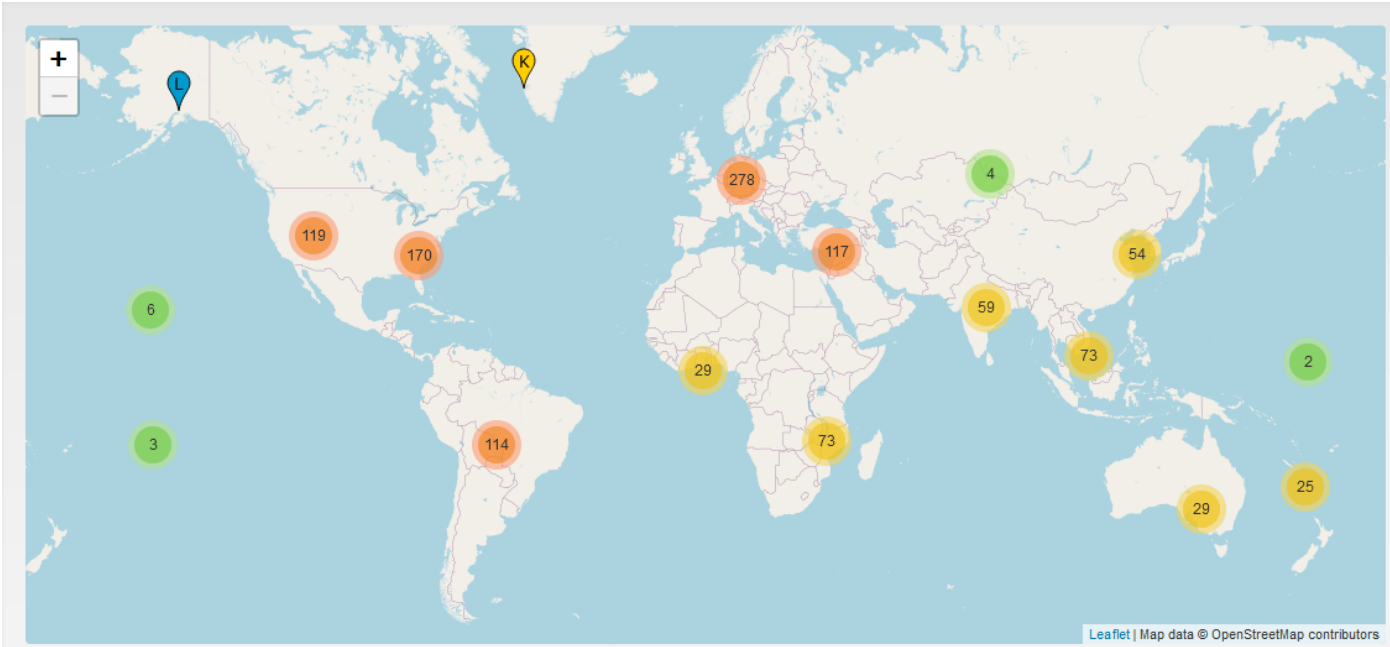
13 root servers (nommés de A à M), utilisation de l'**anycast.**

Voir *Root-servers.org*

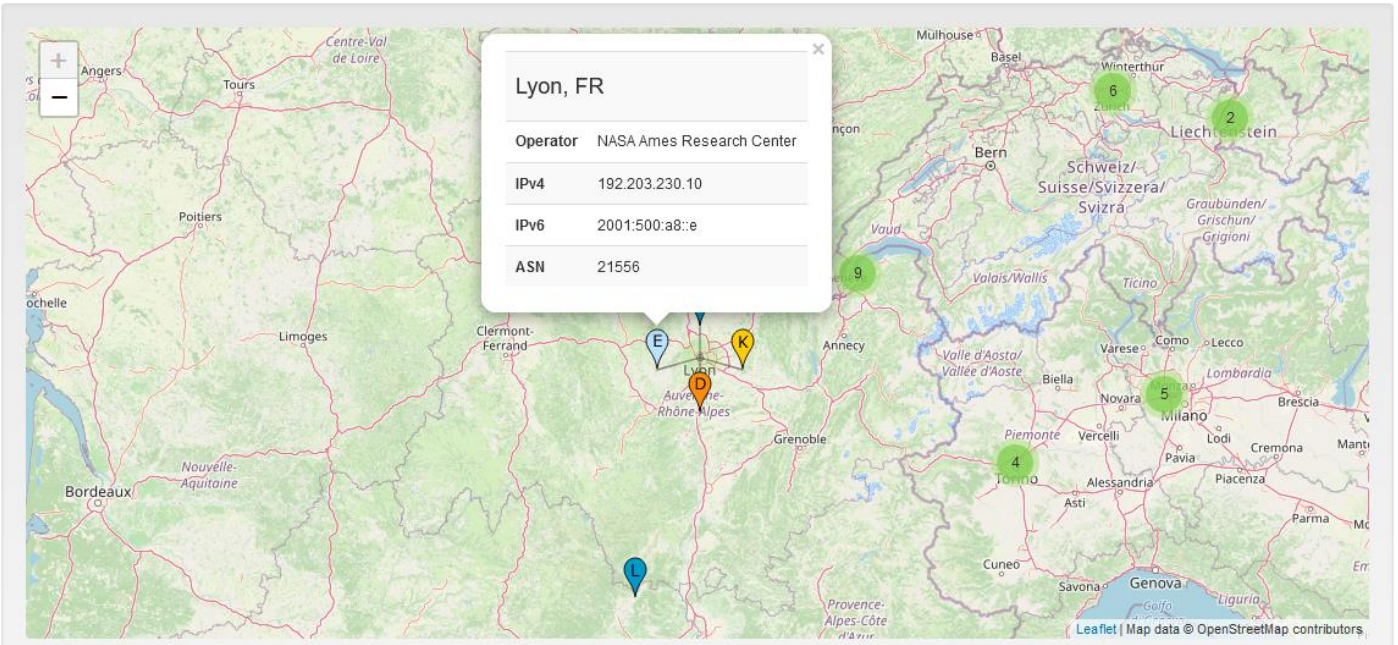
Serveur	Opérateur	Soft
A	Verisign	BIND
B	USC-ISI	BIND
C	Cogent Communications	BIND
D	University of Maryland	BIND
E	NASA	BIND
F	Internet Systems Consortium	BIND 9[15]
G	Defense Information Systems Agency	BIND
H	U.S. Army Research Lab	NSD
I	Netnod	BIND
J	Verisign	BIND
K	RIPE NCC	NSD[22]
L	ICANN	NSD[26]
M	WIDE Project	BIND

- **www.isc.org → F-server, 46 nœuds, 16k.hits/s (par nœud)**
- ripe-ncc → **K-server, 5 à 9 k.hits/s**
- l'ISC (Internet System Consortium) gère également le logiciel bind

Pourquoi 13 serveurs root ?



As of 03/17/2021 9:06 a.m., the root server system consists of 1375 instances operated by the 12 independent root server operators.



As of 03/17/2021 9:06 a.m., the root server system consists of 1375 instances operated by the 12 independent root server operators.

Source : isc.org

Implémentation (resolver) :

Root Files

Root Hints

Operators who manage a DNS recursive resolver typically need to configure a *"root hints file"*. This file contains the names and IP addresses of the authoritative name servers for the root zone, so the software can bootstrap the DNS resolution process. For many pieces of software, this list comes built into the software.

- Root Hints File (FTP)
- Root Hints File (HTTP)

```
;       This file holds the information on root name servers needed to
;       initialize cache of Internet domain name servers
;       (e.g. reference this file in the "cache  .  <file>"
;       configuration  file of BIND domain name servers).
;
;       This file is made available by InterNIC
;       under anonymous FTP as
;           file                /domain/named.cache
;           on server           FTP.INTERNIC.NET
;       -OR-                    RS.INTERNIC.NET
;
;       last update:    February 17, 2021
;       related version of root zone:     2021021701
;
; FORMERLY NS.INTERNIC.NET
;
.                       3600000      NS    A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET.     3600000      A     198.41.0.4
A.ROOT-SERVERS.NET.     3600000      AAAA  2001:503:ba3e::2:30
;
; FORMERLY NS1.ISI.EDU
;
.                       3600000      NS    B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET.     3600000      A     199.9.14.201
B.ROOT-SERVERS.NET.     3600000      AAAA  2001:500:200::b
;
.
.
.

; FORMERLY C.PSI.NET
;
.                       3600000      NS    C.ROOT-SERVERS.NET.
;
.                       3600000      NS    G.ROOT-SERVERS.NET.
;
; OPERATED BY RIPE NCC
;
L.ROOT-SERVERS.NET.     3600000      A     199.7.83.42
L.ROOT-SERVERS.NET.     3600000      AAAA  2001:500:9f::42
;
; OPERATED BY WIDE
;
.                       3600000      NS    M.ROOT-SERVERS.NET.
M.ROOT-SERVERS.NET.     3600000      A     202.12.27.33
M.ROOT-SERVERS.NET.     3600000      AAAA  2001:dc3::35
; End of file
```

6. Autour de DNS

- DynDsns → RFC2136, dyndns.org ou dyndns.fr
- DNSSEC → RFC 2535/4033
    - Hors de portée de ce cours
    - A retenir : DNSSEC ne garantit **que** l'identité de l'origine de la réponse.
- Alternative DNS root (DOT 2.0)
- Split DNS (bind)
- GeoDNS (bind)
- Autres utilisations, exemple DKIM :

le serveur DNS autoritaire sur une zone publie dans des RR TXT des clé publiques qui servent à vérifier la signature de certains messages

Implémentations principales :

- **BIND** (isc.org) dernière version stable[12] : v9.18.1
- **NSD** (open source) : 3 root servers
- **Microsoft DNS**

[12] *Au 23 mars 2022*

7. La sécurité

- **DNS** Indispensable pour presque toutes les applications ⇒ point critique (**SPOF**)
- Données potentiellement sensibles (plus qu'il ne peut y paraître[14])
- Service vulnérable[13] :
    - Confidentialité[14] : pas d'authentifications et les données circulent en clair
    - Intégrité : les messages du protocole ne sont pas (toujours) signés
    - Disponibilité : DNS vulnérable aux attaques DOS/DDOS[15]

Causes principales de ces vulnérabilités :

- **L'emploi d'UDP**
- **L'absence de fonctions d'authentification et de chiffrement dans le protocole**

Sur un autre plan, contrôle du système de nommage par des autorités aux intentions diverses :

- Juridique / propriété industrielle (fiabilité des « noms »)
- Surveillance : NSA programme *MoreCowBell*[16], lois sur le renseignement, etc.

[13] *RFC 3833 (2004) : Threat Analysis of the Domain Name System (DNS)*
[14] *RFC 7626 DNS Privacy : 1er RFC publié par Bortzmeyer !*
[15] *Attaque Mirai / Dyn. Oct. 2016*
[16] *Le programme MORECOWBELL de la NSA sonne le glas du DNS - hal.archives-ouvertes.fr/hal-01114307/*

DSN menteurs :

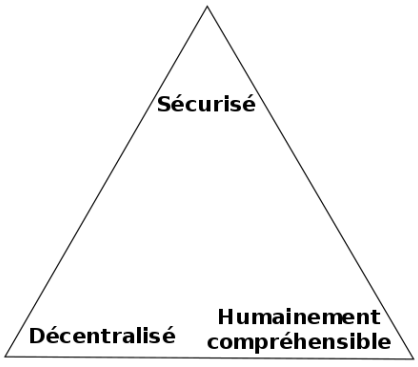
- Les serveurs récursifs « gratuits »
- Les serveurs des FAI
  - Pour des obligations « légales », par ex. sécurité intérieure[17]
  - Pour des raison « commerciales » (censure des publicités pour un concurrent par exemple)
- Les « portails captifs »

  <Rappel du principe>
  - Pour des obligations « légales » : traçabilité des connexions[18]

  (est-on obligé d'avoir un portail captif à son domicile ?)

Atténuations (Mitigation) :

1. Architecture : placer le serveur récursif « au plus près » des utilisateurs (confiance)
2. Améliorations des implémentations
3. DNS-SEC (difficile à mettre en place, et n'assure pas toutes les protections)
4. Projets en cours : T-DNS (DNS sur DTLS), DnsCurve, INS, Namecoin,… Mais difficultés pour énoncer les propriétés d'un 'bon' système de nommage (**triangle de Zooko**)



[17] *Bortzmeier 2009 : Censure administrative du Web en France, un premier regard technique*

[18] *« Grenoble : des gérants de bar placés en garde à vue à cause de leur wifi ». Le Dauphiné, 1er oct. 2020*

8. Ce qu'il faut savoir faire

Client :

- Configurer la gestion des noms
  - Unix : ***resolv.conf***
  - Windows : chercher
  - Tester la résolution de noms (quel serveur, pourquoi etc.)
  - Installer un serveur récursif autonome : par exemple *unbound*[19], vérifier ;
- Tester, dépanner
  - **dig , nslookup**
  - Vider le cache :
    - windows : ***#ipconfig /flushdns***
    - unix (bind) : ***/etc/init.d/named restart***
    - navigateurs (Firefox, par ex.)

Serveur :

- Faire connaissance avec bind (isc.org) : versions, doc officielles.
- Installer complètement un serveur bind sous Linux (depuis des sources vérifiées)
- Faire une configuration minimale
  - Serveur autorité (SOA)
  - Résolution (A)
  - Délégation (NS + Glue record)
  - Messagerie (MX)
  - Résolution inverse (PTR, « in-addr-arpa »)

  Préparer TP N°1

[19] *https://nlnetlabs.nl/projects/unbound/about/*