Système de nommage sur Internet — DNS

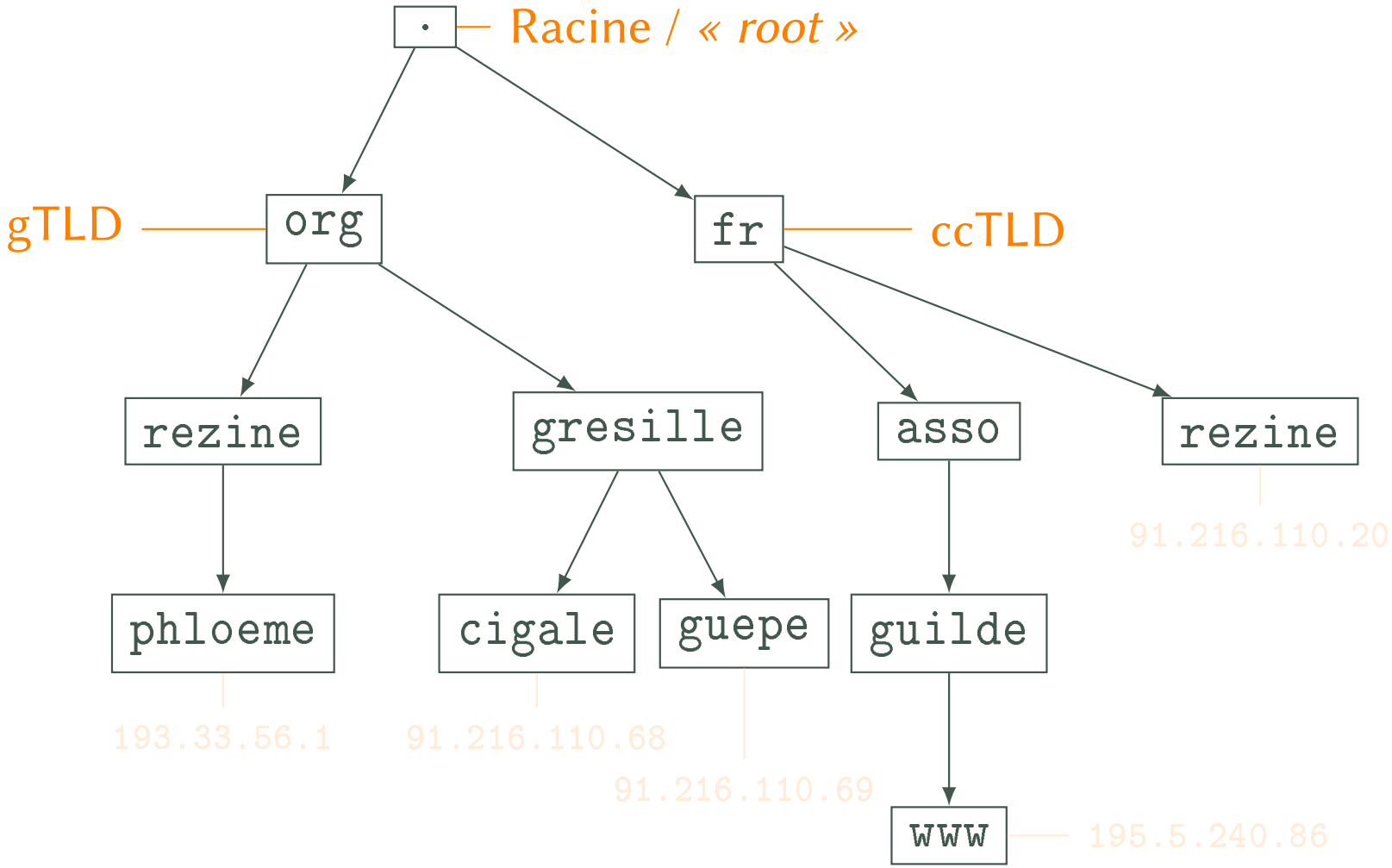
Thomas vO

Grésille

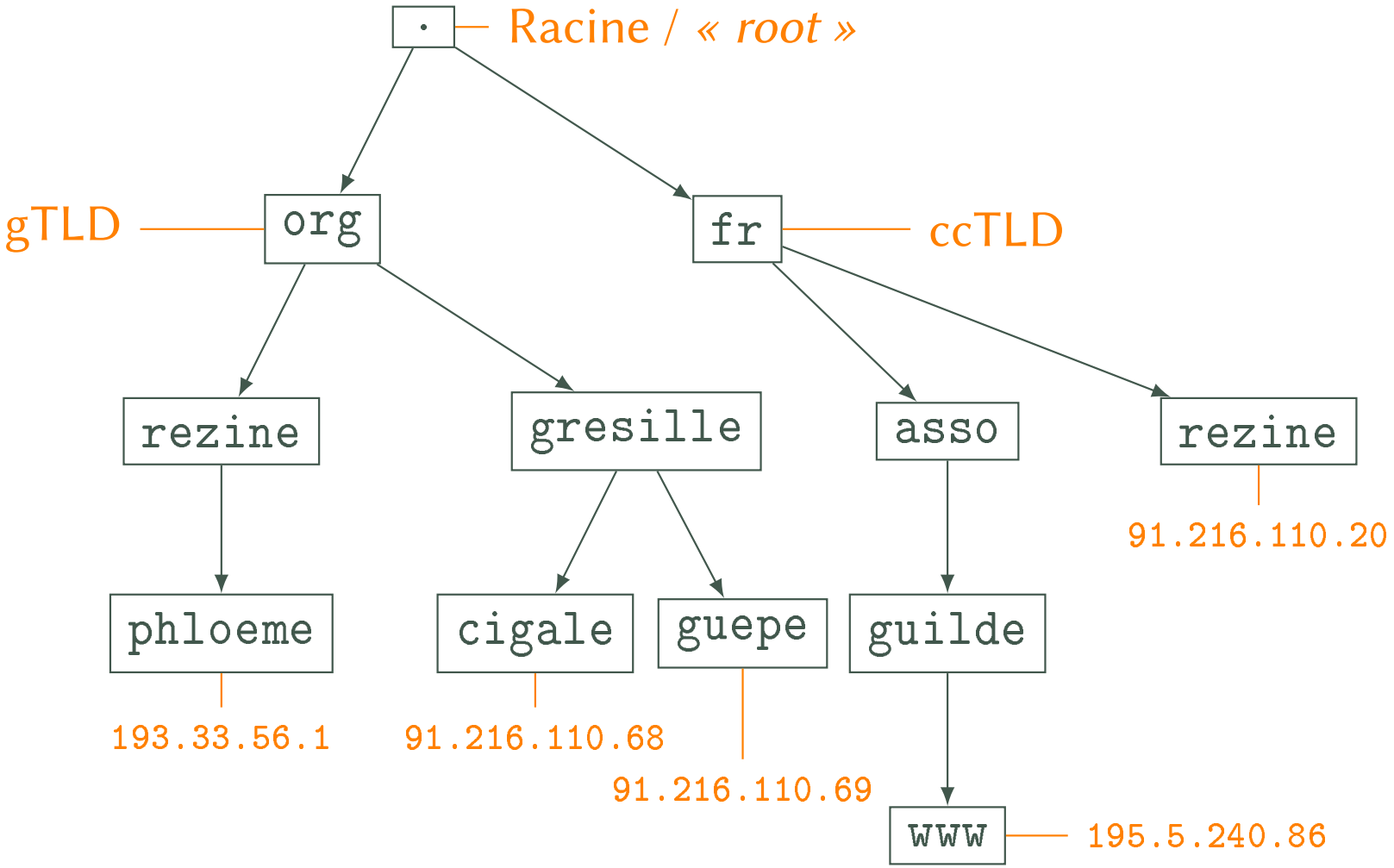
1er mars 2016

Plan

L'arborescence des noms de domaine



L'arborescence des noms de domaine



Historique

Le DNS (*Domain Name System*) est un protocole permettant de résoudre des noms en données.

- Avant 1982, un fichier `hosts.txt` (RFC608) était maintenu sur chaque système
- En 1983, le design du système DNS est publié dans les RFC882 et RFC883
- En 1987, la norme DNS est codifiée dans les RFC1034 et RFC1035
- En 1994, la RFC1591 normalise les ccTLDs
- En 2003, le format « *Punycode* » est créé par la RFC3490, permettant l'internationalisation des noms de domaine :
  - www.potamochère.fr
  - www.académie-française.fr
  - .中国 est le TLD chinois (`.cn`)

Historique

Le DNS (*Domain Name System*) est un protocole permettant de résoudre des noms en données.

- Avant 1982, un fichier `hosts.txt` (RFC608) était maintenu sur chaque système
- En 1983, le design du système DNS est publié dans les RFC882 et RFC883
- En 1987, la norme DNS est codifiée dans les RFC1034 et RFC1035
- En 1994, la RFC1591 normalise les ccTLDs
- En 2003, le format « *Punycode* » est créé par la RFC3490, permettant l'internationalisation des noms de domaine :
  - www.potamochère.fr
  - www.académie-française.fr
  - .中国 est le TLD chinois (`.cn`)

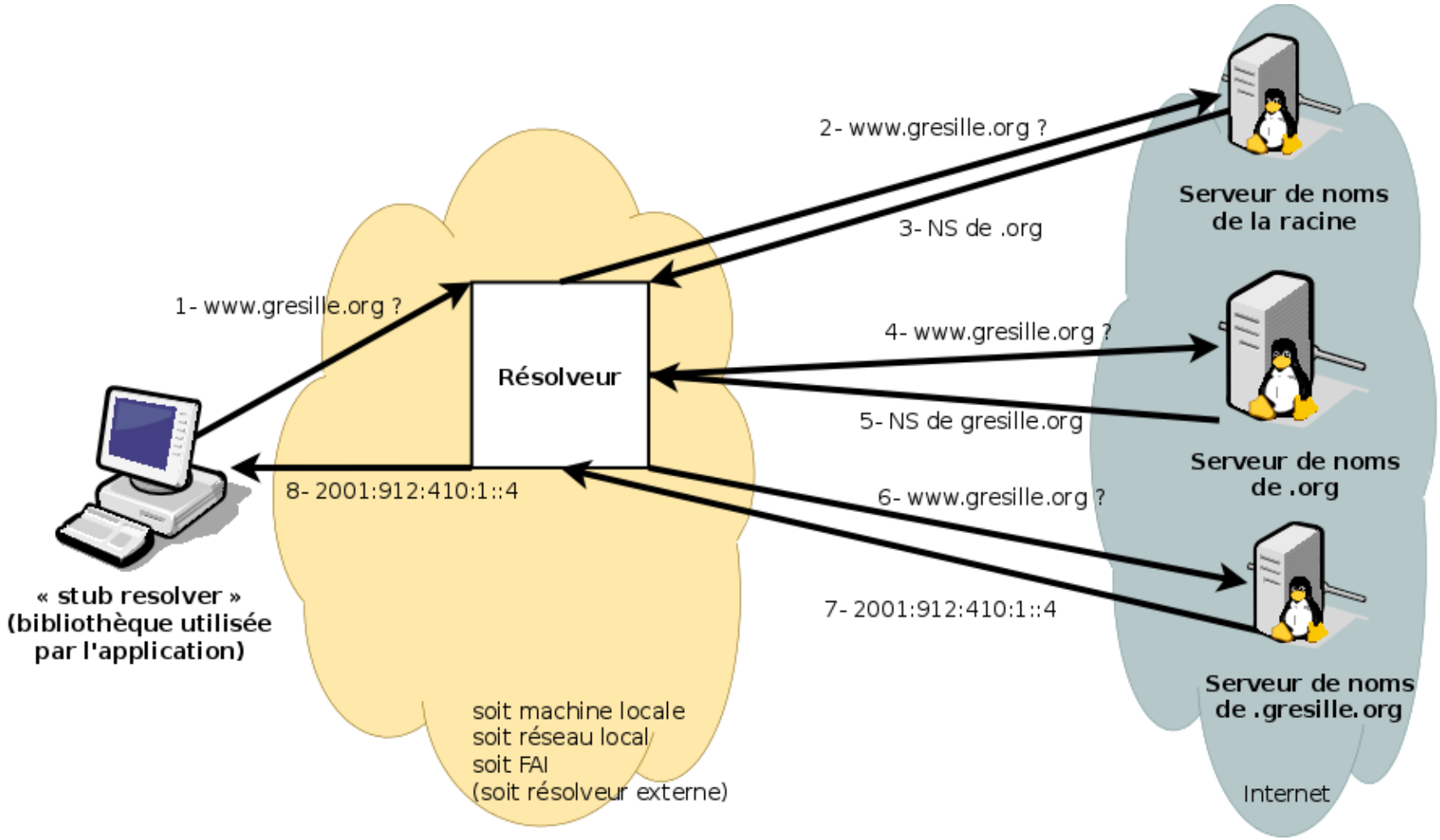
Historique

Le DNS (*Domain Name System*) est un protocole permettant de résoudre des noms en données.

- Avant 1982, un fichier `hosts.txt` (RFC608) était maintenu sur chaque système
- En 1983, le design du système DNS est publié dans les RFC882 et RFC883
- En 1987, la norme DNS est codifiée dans les RFC1034 et RFC1035
- En 1994, la RFC1591 normalise les ccTLDs
- En 2003, le format « *Punycode* » est créé par la RFC3490, permettant l'internationalisation des noms de domaine :
  - www.potamochère.fr
  - www.académie-française.fr
  - .中国 est le TLD chinois (`.cn`)

Plan

Résolveur ou serveur récursif (1)



Résolveur ou serveur récursif (2)

Un serveur récursif sert à répondre aux requêtes de clients à propos de n'importe quel nom (généralement sur un réseau précis).
La seule chose dont il a besoin, c'est une liste des IPs des serveurs de la Racine.

Apprendre par cœur l'IP d'un résolveur DNS ouvert est une bonne idée

- FDN : `2001:910:800::12` (`80.67.169.12`) et `2001:910:800::40` (`80.67.169.40`)
- LDN : `2001:913::8` (`80.67.188.188`)
- Google : `2001:4860:4860::8888` (`8.8.8.8`) et `2001:4860:4860::8844` (`8.8.4.4`)

Résolveur ou serveur récursif (2)

Un serveur récursif sert à répondre aux requêtes de clients à propos de n'importe quel nom (généralement sur un réseau précis).
La seule chose dont il a besoin, c'est une liste des IPs des serveurs de la Racine.

Apprendre par cœur l'IP d'un résolveur DNS ouvert est une bonne idée

- FDN : `2001:910:800::12` (`80.67.169.12`) et `2001:910:800::40` (`80.67.169.40`)
- LDN : `2001:913::8` (`80.67.188.188`)
- Google : `2001:4860:4860::8888` (`8.8.8.8`) et `2001:4860:4860::8844` (`8.8.4.4`)

Résolveur ou serveur récursif (2)

Un serveur récursif sert à répondre aux requêtes de clients à propos de n'importe quel nom (généralement sur un réseau précis).
La seule chose dont il a besoin, c'est une liste des IPs des serveurs de la Racine.

Apprendre par cœur l'IP d'un résolveur DNS ouvert est une bonne idée

- FDN : `2001:910:800::12` (`80.67.169.12`) et `2001:910:800::40` (`80.67.169.40`)
- LDN : `2001:913::8` (`80.67.188.188`)
- Google : `2001:4860:4860::8888` (`8.8.8.8`) et `2001:4860:4860::8844` (`8.8.4.4`)

Serveur faisant autorité (1)

Un serveur faisant autorité fait autorité sur sa zone.

- Les serveurs de la racine connaissent les serveurs des TLD
- Les serveurs d'un TLD connaissent tous les noms de premier niveau sous ce TLD
- Et ainsi de suite…

Un serveur peut ainsi *déléguer* un domaine à un autre serveur.

On utilise pour ça des « *glue records* », qui sont une redondance d'information dans la zone mère et la zone fille (les enregistrements NS et leurs IP associées).

Serveur faisant autorité (1)

Un serveur faisant autorité fait autorité sur sa zone.

- Les serveurs de la racine connaissent les serveurs des TLD
- Les serveurs d'un TLD connaissent tous les noms de premier niveau sous ce TLD
- Et ainsi de suite…

Un serveur peut ainsi *déléguer* un domaine à un autre serveur.

On utilise pour ça des « *glue records* », qui sont une redondance d'information dans la zone mère et la zone fille (les enregistrements NS et leurs IP associées).

Serveur faisant autorité (1)

Un serveur faisant autorité fait autorité sur sa zone.

- Les serveurs de la racine connaissent les serveurs des TLD
- Les serveurs d'un TLD connaissent tous les noms de premier niveau sous ce TLD
- Et ainsi de suite…

Un serveur peut ainsi *déléguer* un domaine à un autre serveur.

On utilise pour ça des *« glue records »*, qui sont une redondance d'information dans la zone mère et la zone fille (les enregistrements NS et leurs IP associées).

Serveur faisant autorité (2)

On distingue généralement les serveurs maîtres et esclaves (ou primaires et secondaires).

- Un serveur maître a une copie *originale* des données
- Un serveur esclave récupère ses données sur un serveur maître, via des systèmes de notification et de transfert de zones

Pour une zone :

- Il existe au moins un serveur maître
- Il peut exister plusieurs serveurs esclaves
- Le(s) serveur(s) maître(s) peuvent être accessibles publiquement, ou uniquement aux serveurs esclaves, qui se chargeront de répondre à Internet

Serveur faisant autorité (2)

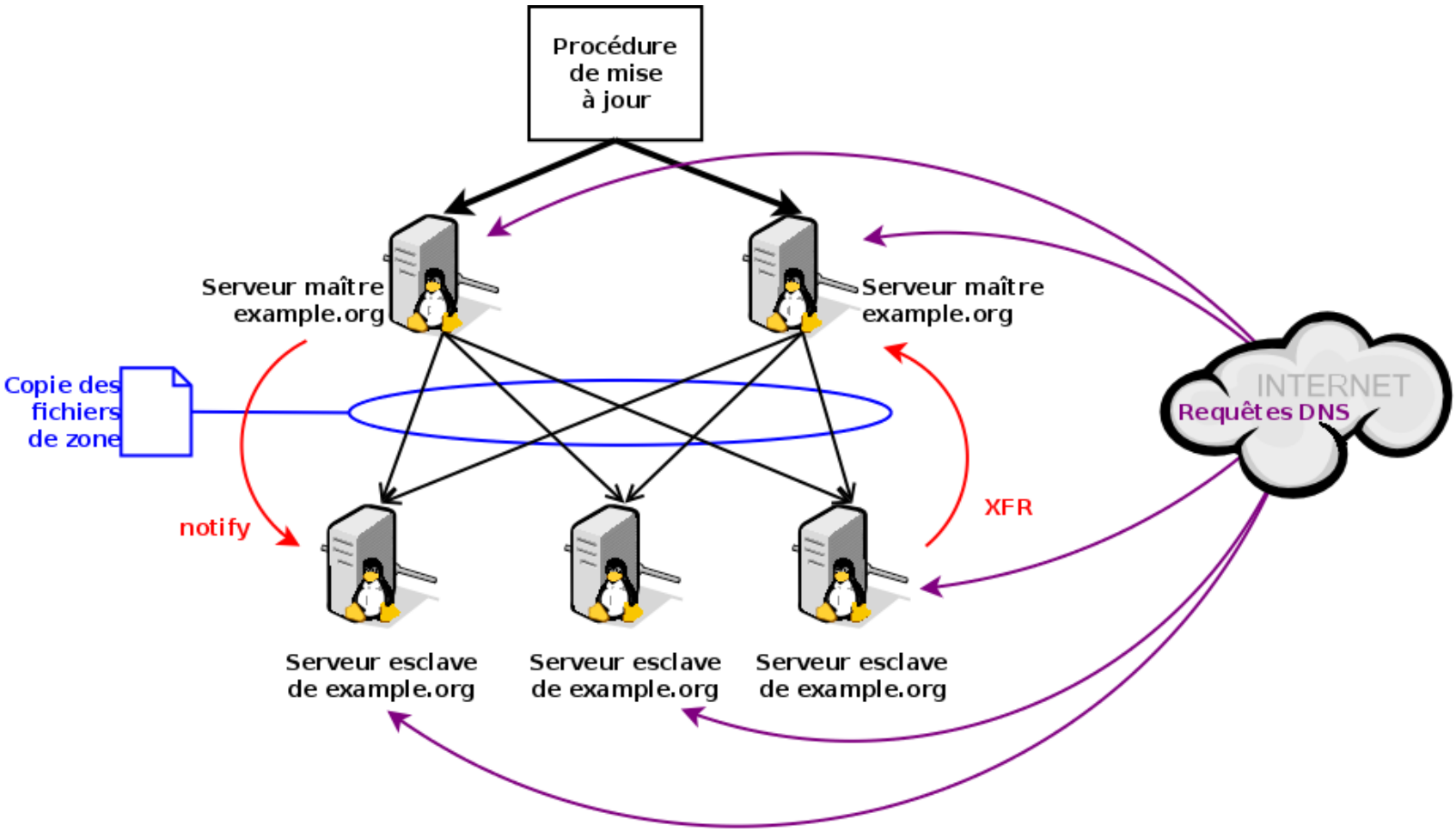
On distingue généralement les serveurs maîtres et esclaves (ou primaires et secondaires).

- Un serveur maître a une copie *originale* des données
- Un serveur esclave récupère ses données sur un serveur maître, via des systèmes de notification et de transfert de zones

Pour une zone :

- Il existe au moins un serveur maître
- Il peut exister plusieurs serveurs esclaves
- Le(s) serveur(s) maître(s) peuvent être accessibles publiquement, ou uniquement aux serveurs esclaves, qui se chargeront de répondre à Internet

Serveurs faisant autorité (3)



Fichiers de zone — généralités (1)

Le format général des enregistrements DNS est :
`<nom> [TTL] <classe> <type> <valeur>`

- `nom` : l'enregistrement que l'on souhaite voir résoudre
- `TTL` : « *Time To Live* » la durée à conserver pour les résolveurs
- `classe` : `IN` pour Internet (`CH` pour Chaos System et `HS` pour Hesiod)
- `type` : détaillé plus loin
- `valeur` : la valeur à retourner pour cet enregistrement

Fichiers de zone — généralités (1)

Le format général des enregistrements DNS est :
`<nom> [TTL] <classe> <type> <valeur>`

- `nom` : l'enregistrement que l'on souhaite voir résoudre
- `TTL` : « *Time To Live* » la durée à conserver pour les résolveurs
- `classe` : `IN` pour Internet (`CH` pour Chaos System et `HS` pour Hesiod)
- `type` : détaillé plus loin
- `valeur` : la valeur à retourner pour cet enregistrement

Fichiers de zone — types d'enregistrements (2)

- `NS` désigne les serveurs de noms
- `A` désigne l'adresse IPv4
- `AAAA` désigne l'adresse IPv6
- `MX` désigne les serveurs de mails (assortis d'une priorité)
- `CNAME` désigne un alias (et `DNAME` pour les sous-noms)
- `TXT` désigne du texte
- `SRV` désigne un enregistrement serveur
- `PTR` sert pour les enregistrements inverses (zones `in-addr.arpa.` et `ip6.arpa.`)
- `HINFO` désigne des informations (CPU, OS) d'un équipement

Fichiers de zone — syntaxe (3)

- ; est un commentaire
- `$ORIGIN` permet de fixer un suffixe
- `@` désigne le nom défini avec `$ORIGIN`
- `SOA` : « *Start Of Authority* », est la première information que l'on doit définir pour une zone :

Début de zone

```
@   IN  SOA ns2.gresille.org. root.gresille.org. (
        2016010301  ; Serial
        3600        ; Refresh
        900         ; Retry
        2419200     ; Expire
        3600 )      ; Negative Cache TTL
```

Fichiers de zone — syntaxe (3)

- ; est un commentaire
- `$ORIGIN` permet de fixer un suffixe
- `@` désigne le nom défini avec `$ORIGIN`
- `SOA` : « *Start Of Authority* », est la première information que l'on doit définir pour une zone :

Début de zone

```
@    IN   SOA ns2.gresille.org. root.gresille.org. (
          2016010301   ; Serial
          3600         ; Refresh
          900          ; Retry
          2419200      ; Expire
          3600 )       ; Negative Cache TTL
```

Fichiers de zone — exemple (4)

Exemples d'enregistrements

```
; enregistrement SOA - cf. plus haut

@         IN   NS       ns2.gresille.org.
@         IN   NS       ns5.grenode.net.
@         IN   MX   10  mx.gresille.org.
@         IN   MX   20  mx1.grenode.net.
@         IN   TXT      "v=spf1 a mx ?all"

@         IN   A        91.216.110.69
mx        IN   A        91.216.110.67
mx        IN   AAAA     2001:912:410:1::3
web       IN   A        91.216.110.112
*.web     IN   CNAME    web

_jabber._tcp        IN SRV  5 0 5269 chenille.gresille.org.
_xmpp-server._tcp   IN SRV  5 0 5269 chenille.gresille.org.
_xmpp-client._tcp   IN SRV  5 0 5222 chenille.gresille.org.
```

Fichiers de zone — exemple enregistrements reverse (5)

Zone reverse IPv6 (2001:912:410::0)

```
; enregistrements SOA et NS
$ORIGIN 0.0.0.0.0.0.0.0.0.0.1.4.0.2.1.9.0.1.0.0.2.ip6.arpa.
; 2001:912:410:0:0:0:0:67 :
7.6.0.0.0.0.0.0.0.0.0.0.0.0.0.0    IN   PTR      fourmi.gresille.org.
; 2001:912:410:0:0:0:0:68 :
8.6.0.0.0.0.0.0.0.0.0.0.0.0.0.0    IN   PTR      cigale.gresille.org.
```

Zone reverse IPv4 (91.216.110.0)

```
; enregistrements SOA et NS
$ORIGIN 110.216.91.in-addr.arpa.
; 91.216.110.67 :
67        IN     PTR     fourmi.gresille.org.
; 91.216.110.68 :
68        IN     PTR     cigale.gresille.org.
```

Quelques outils

Noms de quelques logiciels utiles quand on joue avec le DNS :

- Pour faire des requêtes DNS : `host`, `nslookup`, `dig`
- Serveurs récursifs : `bind`, `unbound`, `dnsmasq`
- Serveurs faisant autorité : `bind`, `nsd`, `knot`
- Vérifications : `named-checkconf`, `named-checkzone`, `check-soa`, zonemaster

Plan

Le système actuel de nommage sur Internet

Actuellement, le système de nommage propose des identifiants qui ont ces caractéristiques :

- humainement « interprétables » ✓ `www.gresille.org` par rapport à `2001:912:410:1::4`
- globalement uniques ✓ allocation décentralisée
- stables (dans le temps) ∼ relativement stables
- sûrs (niveau juridique, financier, technique,...) ✗ des risques
- résolvables (système de transformation pour accéder à la ressource) ✓ le DNS
- enregistrables facilement (procédures,...) ✗ prix, bureaux d'enregistrement,...

A priori, il n'est pas possible d'avoir toutes ces qualités à la fois.

Gestion décentralisée

Les noms de domaines sont décentralisés. Ainsi :

- Pour avoir un nom dans `.`, l'ICANN fixe les règles (180 000 $)
- Pour avoir un nom dans `.fr`, c'est l'AFNIC qui fixe les règles
- Pour avoir un nom dans `.asso.fr`, c'est l'AFNIC qui fixe les règles
- Pour avoir un nom dans `guilde.asso.fr`, c'est la Guilde qui décide

Enregistrement

- Pour la plupart des TLD, il faut passer par un bureau d'enregistrement
- Le bureau d'enregistrement fait le lien avec le registre
- Par exemple pour `rezine.fr`, le bureau d'enregistrement est Gandi, qui fait le lien avec le registre de `.fr`, l'AFNIC
- Le gouvernement US a mis la main sur la racine (`.`) et en délègue la gestion à l'ICANN (depuis 1998), qui délègue la gestion des TLD à des sociétés
- Certains TLD ont des conditions (très) restrictives

Qui paye ?

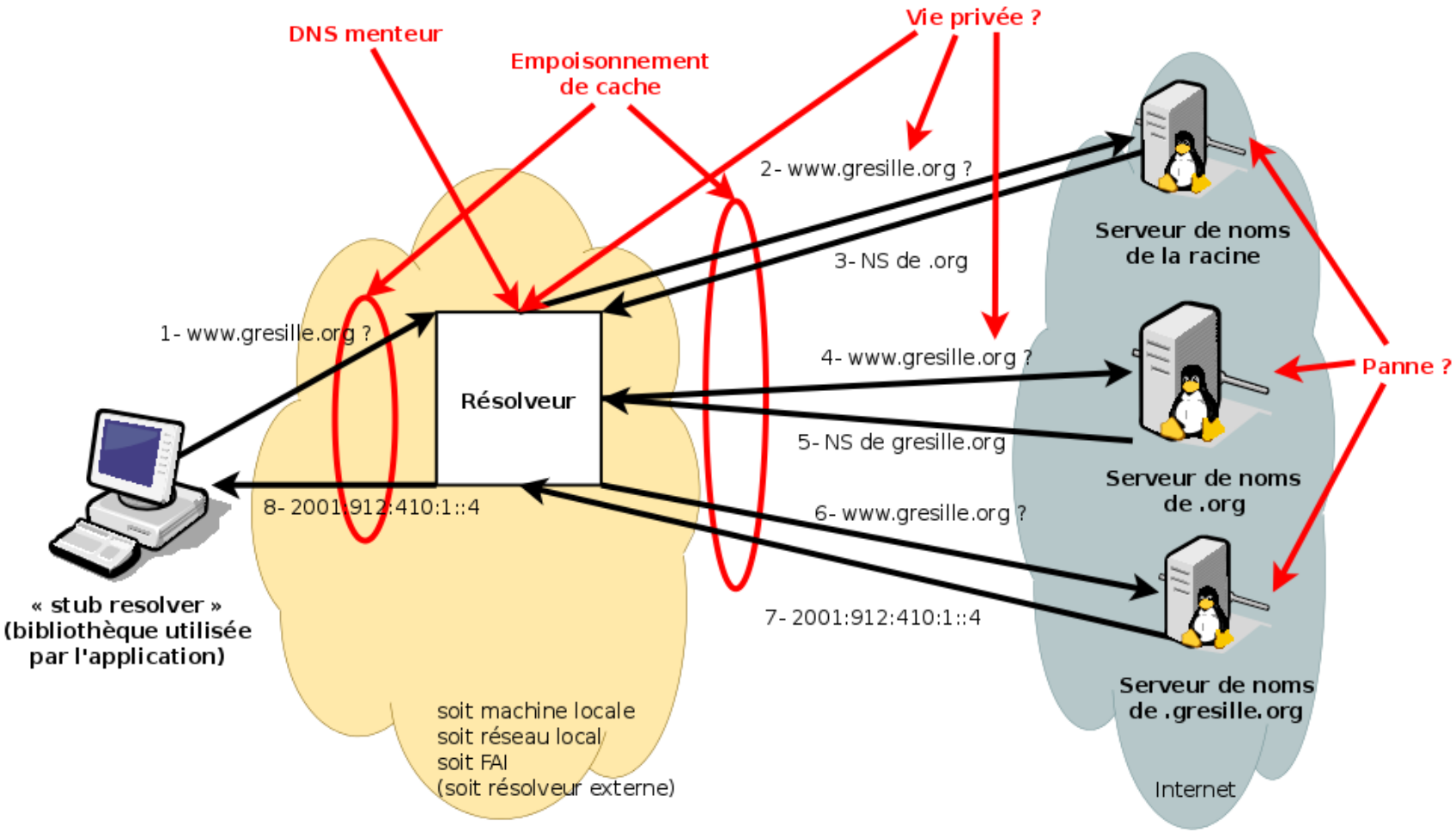
Le DNS est une technologie d'infrastructure, nécessaire mais « invisible ».

- Chaque structure paye son (ses) résolveur(s)
- Chaque structure paye la plupart de ses serveurs faisant autorité (ou échange de bons procédés)
- Depuis 1995, les noms de domaine sont payants
- Les prix d'enregistrement payent des morceaux des serveurs de la Racine
- Les prix d'enregistrement payent des morceaux (la totalité ?) des frais des bureaux d'enregistrement et des registres
- Pour le reste, c'est assez flou (par exemple, l'AFNIC : $\sim$ 15 M€ de CA, $\sim$ 3 millions de noms (`.fr`) et $\sim$ 3,5 € de coût de production annoncé (pour le `.fr`))

Qui a du pouvoir ?

- l'ICANN (dépend de la loi États-Unienne / Californienne) :
  - Peut-on avoir une racine alternative ? (.42, *open-root*)
  - Peut-on acentrer le système de nommage ? (*namecoin*)
- les opérateurs de serveurs de la Racine (13 instances réparties sur $\sim$ 150 machines un peu partout dans le monde)
- les opérateurs de registre (dépendent de lois nationales ; les ccTLD sont généralement délégués par le gouvernement, en France à l'AFNIC)
- les bureaux d'enregistrement (tout le monde veut des procédures faciles **et** sûres)
- les fournisseurs d'accès (résolveurs menteurs, écoute du trafic)
- les opérateurs de résolveurs DNS

Risques techniques (1)



Risques techniques (2)

- Pour éviter les DNS menteurs, il est possible d'installer un résolveur sur sa machine, ou d'utiliser des résolveurs de confiance
- Pour éviter les pannes des serveurs de noms, il est conseillé d'avoir des serveurs faisant autorité dans des réseaux différents
- Risques d'empoisonnement de cache (de résolveurs) : DNSSEC
  - Les serveurs faisant autorité diffusent des signatures cryptographiques des enregistrements
  - Les résolveurs, avec les IPs de la racine, ont la clef de la racine
  - Les signatures sont ensuite arborescentes
  - DNSSEC crée les types suivants :
    - `DNSKEY` permet de donner l'empreinte de la clef publique
    - `DS` permet de donner la clef d'une zone fille
    - `NSEC(3)` permet de prouver la véracité de la non-existence d'un enregistrement

Risques techniques (2)

- Pour éviter les DNS menteurs, il est possible d'installer un résolveur sur sa machine, ou d'utiliser des résolveurs de confiance
- Pour éviter les pannes des serveurs de noms, il est conseillé d'avoir des serveurs faisant autorité dans des réseaux différents
- Risques d'empoisonnement de cache (de résolveurs) : DNSSEC
  - Les serveurs faisant autorité diffusent des signatures cryptographiques des enregistrements
  - Les résolveurs, avec les IPs de la racine, ont la clef de la racine
  - Les signatures sont ensuite arborescentes
  - DNSSEC crée les types suivants :
    - `DNSKEY` permet de donner l'empreinte de la clef publique
    - `DS` permet de donner la clef d'une zone fille
    - `NSEC(3)` permet de prouver la véracité de la non-existence d'un enregistrement

Risques techniques (2)

- Pour éviter les DNS menteurs, il est possible d'installer un résolveur sur sa machine, ou d'utiliser des résolveurs de confiance
- Pour éviter les pannes des serveurs de noms, il est conseillé d'avoir des serveurs faisant autorité dans des réseaux différents
- Risques d'empoisonnement de cache (de résolveurs) : DNSSEC
  - Les serveurs faisant autorité diffusent des signatures cryptographiques des enregistrements
  - Les résolveurs, avec les IPs de la racine, ont la clef de la racine
  - Les signatures sont ensuite arborescentes
  - DNSSEC crée les types suivants :
    - `DNSKEY` permet de donner l'empreinte de la clef publique
    - `DS` permet de donner la clef d'une zone fille
    - `NSEC(3)` permet de prouver la véracité de la non-existence d'un enregistrement

Sources

- Stéphane Bortzmeyer : son blog et ses conférences, liste non exhaustive :
  - Il était une fois Internet
  - Sécurité du DNS : DNSSEC (JRES 2009)
  - Des clefs dans le DNS, un successeur à X509 (JRES 2011)
  - Persée et la Gorgone : attaques par déni de service utilisant le DNS, et les contre-mesures (JRES 2013)
- Benjamin Bayart, conférence DNS et Mail
- Différentes pages de Wikipédia, francophone et anglophone
- Les RFC
- Le site web de l'AFNIC