



DNS Session 1: Principes de base

Les ordinateurs utilisent des adresses IP. Pourquoi avons nous besoin des noms?

- Faciles aux êtres humains de mémoriser
- Les ordinateurs peuvent être déplacés entre les réseaux, dans ce cas leurs adresses IP changent

Solution ancienne : hosts.txt

- Le fichier est maintenu de façon centralisée et distribué à toutes les machines sur Internet

<i>SPARKY</i>	<i>128.4.13.9</i>
<i>UCB-MAILGATE</i>	<i>4.98.133.7</i>
<i>FTPHOST</i>	<i>200.10.194.33</i>
<i>... etc</i>	

Cette rubrique existe encore:
/etc/hosts [Unix]
c:\windows\hosts [Windows]

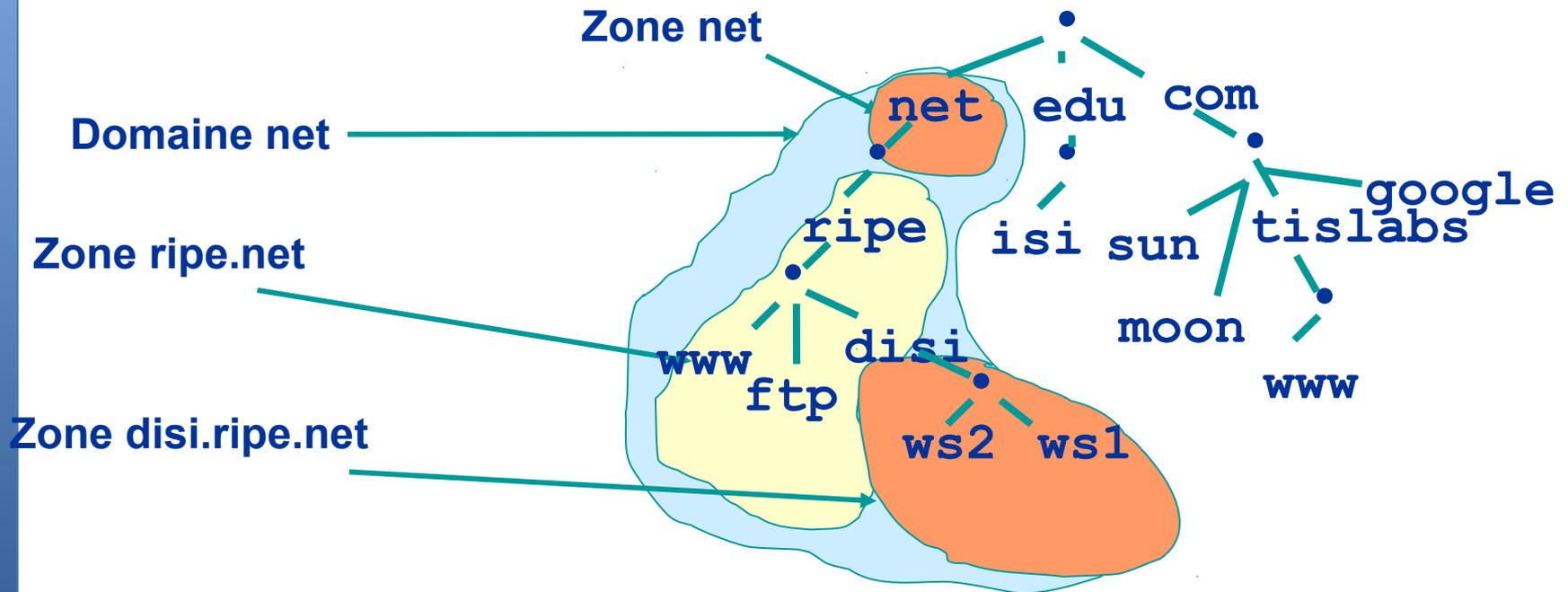
Hosts.txt est inadapté à grande échelle

- Fichier volumineux
- Nécessite d'être copié fréquemment sur tous les hôtes
- Uniformité
- toujours dépassé
- Unicité de nom
- Un seul point d'administration

Le Domain Name System est née

- Le DNS est une base de données **distribuée** qui fait correspondre le nom à une adresse IP (et à d'autres informations)
- Distribuée:
 - Partage l'administration
 - Partage la charge
- **robustesse et performance à travers :**
 - **La réplication**
 - **Le système cache**
- Une pièce ***critique*** de l'infrastructure Internet

DNS est hiérarchique (1)



Forme la structure de l'arborescence

DNS est hiérarchique (2)

- Donne globalement des noms uniques
- Administration par zones (des parties de l'arborescence)
- Vous pouvez donner (" délégué ") le contrôle d'une partie de l'arborescence sous vous
- Exemple:
 - .net est sur un ensemble de serveurs de noms
 - ripe.net est sur un ensemble différent
 - dissi.ripe.net est sur un autre ensemble

Les noms de domaine sont (presque) illimités

- Longueur totale de 255 caractères maximum
- Dans chaque partie 63 caractères Maximum(RFC 1034, RFC 1035)
- Si le nom de domaine est utilisé comme un nom d'hôte, vous devez respecter quelques restrictions
 - RFC 952 (dépassé !)
 - a-z 0-9 et tiret (-)uniquement
 - Pas de underscores (_)

UTILISATION DU DNS

- Un nom de domaine (comme `www.tiscali.co.uk`) est une CLEF pour rechercher une information
- Le résultat est un ou plusieurs Enregistrements de Ressources (ER)
- Il y a des ER différents pour différents types d'information
- Vous pouvez demander le type spécifique que vous voulez, ou demandez " n'importe quel " ER associé au nom de domaine

VUE GENERALE DES ERs

- **A** (adresse): associe le nom d'hôte à l'adresse IP
- **PTR** (pointer): associe l'adresse IP au nom
- **MX** (Mail eXchanger): où délivré le courrier pour l'adresse user@domain
- **CNAME** (Canonical NAME): associe un nom alternatif au nom réel de l'hôte
- **TXT (text)**: tout texte descriptif
- **NS** (Name Server), **SOA** (Start Of Authority): sont utilisés pour la délégation et la gestion du DNS lui-même

Exemple Simple

- REQUETE: `www.tiscali.co.uk`
- TYPE: `A`
- Resultat:

`www.tiscali.co.uk. IN A 212.74.101.10`

Dans ce cas-ci, juste un ER est trouvé,
mais en général, de ER multiples peuvent être retournés

IN est la “class” d’INTERNET, utilisé par DNS

Résultats Possibles

- Positif (un ou plusieurs ER sont trouvés)
- Négatif (certainement aucun ER ne correspond à la requête)
- Échec de serveur (ne peut pas trouver la réponse)

Comment utilisez une adresse IP comme la clef pour une requête DNS?

- Convertir l'adresse IP au format 4 digits séparé par des points
- Renverser les quatre parties
- Ajouter " in-addr.arpa. " à la fin
- domaine spécial réservé à cette fin

Exemple pour chercher le nom de
212.74.101.10 **10.101.74.212.in-addr.arpa.** est PTR
www.tiscali.co.uk

Connue comme " une consultation inverse de DNS "
Parce que nous recherchons le nom pour une adresse IP,
plutôt que l'adresse IP pour un nom)

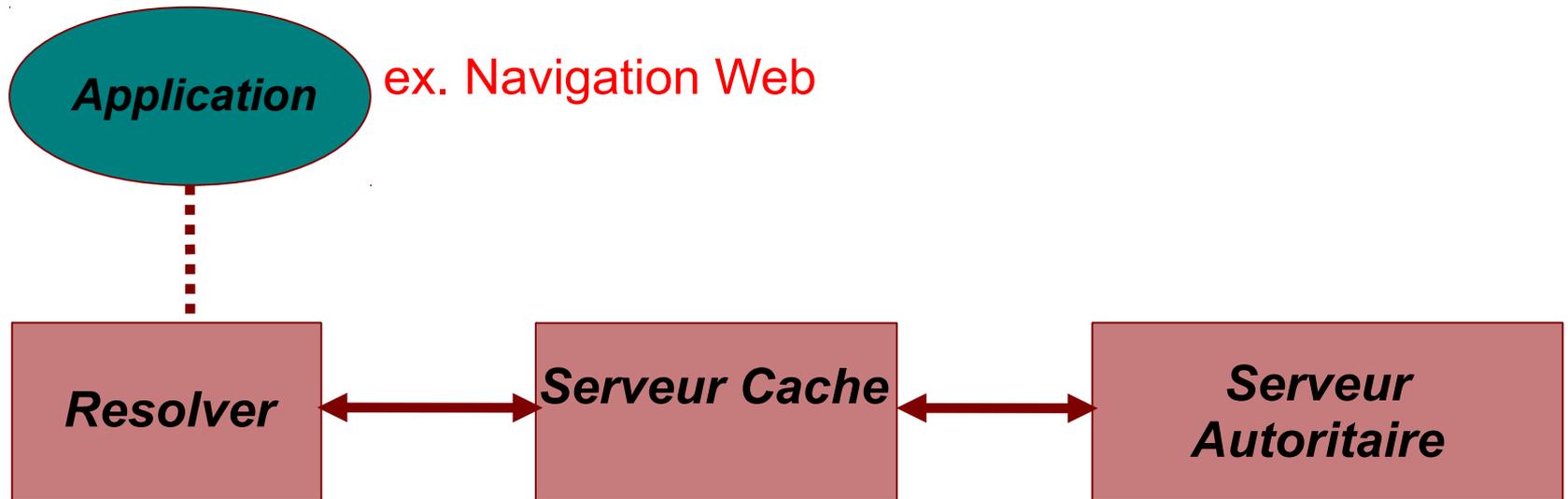


A large, bold, black question mark is centered on a white background. To the left of the page, there is a vertical blue bar that transitions from a lighter blue at the top to a darker blue at the bottom.

Le DNS est une application Client/Serveur

- Naturellement - il fonctionne à travers un réseau
- Les requêtes et les réponses sont normalement envoyées dans des paquets UDP, port 53
- De temps en temps utilise le TCP, port 53
 - pour les requêtes très grandes , exemple : transfert zone à partir du maître à l'esclave

Il y a trois rôles impliqués dans le DNS



Trois rôles du DNS

Le RESOLVER

- prends la demande de l'application,
- formate la demande dans le paquet UDP
- Envoi la demande au cache DNS

SERVEUR CACHE

- Renvoie la réponse si elle est déjà connue
- Autrement il recherche un serveur autoritaire qui a l'information
- Cache le résultat pour de requêtes futures
- Egalement connu sous le nom de **Serveur RECURSIF**

SERVEUR AUTORITAIRE

- Contient l'information réelle mise dans le DNS par le propriétaire du domaine

Trois rôles du DNS

- Le MEME protocole est utilisé pour la communication du resolver au cache et du cache au serveur autoritaire
- Il est possible de configurer un seul serveur de nom en tant que serveur cache et serveur autoritaire à la fois
- Mais il exécute toujours seulement un rôle pour chaque requête entrante
- Ceci est courant mais **NON RECOMMANDÉ** (à voir plus tard)

Rôle 1: LE RESOLVER

- Un morceau de logiciel qui formate une requête DNS dans un paquet UDP, l'envoie à un cache, et décode la réponse
- Habituellement une bibliothèque partagée (ex. libresolv.so sous Unix) parce que beaucoup d'applications ont besoin de lui
- CHAQUE machine a besoin d'un resolver
 - Par exemple: chaque poste de travail Windows en a un

Comment le resolver trouve-t-il le serveur cache?

- Il doit être explicitement configuré (statiquement ou par l'intermédiaire du DHCP, etc...)
- Il doit être configuré avec l'ADRESSE IP du serveur cache (pourquoi pas le nom?)
- Bonne idée de configurer plus d'un cache au cas où le premier tomberait en panne

Comment choisissez-vous quel serveur cache utilisé ?

- Vous devez avoir la **PERMISSION** d'utiliser le serveur cache
 - Ex. serveur cache de votre ISP, ou le vôtre
- **Préférer le serveur cache proche**
 - Réduit au minimum la perte aller-retour de temps et de paquets
 - Peut réduire le trafic sur votre liaison externe, puisque souvent le serveur cache peut répondre sans contacter d'autres serveurs
- **Préférer un serveur cache fiable**
 - Peut-être votre propre serveur cache

Le Resolver peut être configuré avec le(s) domaine(s) par défaut

- Si "foo.bar" échoue, réessayer alors la requête en tant que "foo.bar.mydomain.com"
- Peut sauver la saisie mais ajoute la confusion
- Peut produire du trafic inutile supplémentaire
- A éviter

Exemple: La configuration du resolver de Linux

`/etc/resolv.conf`

search trstech.net

nameserver 196.200.57.6

nameserver 212.74.112.67

C'est tout ce dont vous avez besoin pour configurer un resolver

Les tests du DNS

- Juste saisir dans la zone adresse de votre navigateur : "www.yahoo.com " ?
- Pourquoi est ce que ceci n'est pas un bon test?

Tester le DNS avec "dig"

- "dig" est un programme qui effectue des requêtes DNS et affiche les résultats
- Mieux que "nslookup", "host" parce qu'il montre l'information crue complètement

```
dig tiscali.co.uk.
```

```
-- par défaut pour demander le type  
"A"
```

```
dig tiscali.co.uk. mx
```

```
-- indique le type de requête
```

```
dig @212.74.112.66 tiscali.co.uk. mx
```

```
-- Envoyé à un cache DNS
```

Le point à la fin d'un nom de domaine

dig tiscali.co.uk.

Point final



- Empêche n'importe quel domaine par défaut d'être ajouté
- Prendre l'habitude de l'utiliser au cours des tests du DNS
 - **seulement sur des noms de domaine, pas sur les adresses IP**

```
# dig @81.199.110.100 www.gouv.bj. a
; <<>> DiG 8.3 <<>> @81.199.110.100 www.gouv.bj a
; (1 server found)
;; res options: init recurs defnam dnsrch
;; got answer:
;; ==>>HEADER<<= opcode: QUERY, status: NOERROR, id: 4
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 4, ADD'L: 3
;; QUERY SECTION:
;;      www.gouv.bj, type = A, class = IN

;; ANSWER SECTION:
www.gouv.bj.      1D IN CNAME      waib.gouv.bj.
waib.gouv.bj.    1D IN A           208.164.179.196

;; AUTHORITY SECTION:
gouv.bj.          1D IN NS          rip.psg.com.
gouv.bj.          1D IN NS          ben02.gouv.bj.
gouv.bj.          1D IN NS          nakayo.leland.bj.
gouv.bj.          1D IN NS          ns1.intnet.bj.

;; ADDITIONAL SECTION:
ben02.gouv.bj.    1D IN A           208.164.179.193
nakayo.leland.bj. 1d23h59m59s IN A  208.164.176.1
ns1.intnet.bj.   1d23h59m59s IN A  81.91.225.18

;; Total query time: 2084 msec
;; FROM: ns.tl.ws.afnog.org to SERVER: 81.199.110.100
;; WHEN: Sun Jun  8 21:18:18 2003
;; MSG SIZE  sent: 29  rcvd: 221
```

Interprétation des résultats: header (entête)

➤ STATUS

- NOERROR: 0 ou plus d'ER est retourné
- NXDOMAIN: domaine inexistant
- SERVFAIL: le serveur cache ne pouvait pas localiser la réponse

➤ FLAGS

AA: Réponse de serveurs autoritaires (pas du serveur cache)

Vous pouvez ignorer les autres

QR: Query/Response (1 = Réponse)

RD: Recursion Desired (Résursion Désiré)

RA: Recursion Available (résursion disponible)

Interprétation des résultats

➤ Answer section (Les ERs demandés)

- Chaque enregistrement a un temps de vie (TTL)
- Dit combien de temps le cache le gardera

➤ Authority section

- Quels serveurs de noms sont autoritaires pour ce domaine

➤ Additional section

- Plus d'enregistrements (ERs) : typiquement des adresses IP pour les serveurs de noms autoritaires

➤ Total query time

➤ From

- Indique quel serveur a donné la réponse!
- Si vous faites une faute de frappe, la requête peut aller à un serveur par défaut

Exercices Pratiques

- Configurer le resolver Unix
- Faire des requêtes DNS en utilisant 'dig'
- Utiliser 'tcpdump' pour afficher les requêtes émises qui sont envoyées au cache