

# DNS Session 1 : Principes de base

Arnaud Abdoul Aziz AMELINA

AFNOG 2015, Tunis, Tunisie

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

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

# L'ancienne solution : Hosts.txt

Un fichier est maintenu de facon centralisée et distribué à tous les machines sur Internet

| | |
|---|---|
| • ***SPARKY*** | ***128.4.13.9*** |
| • ***UCB-MAILGATE*** | ***4.98.133.7*** |
| • ***FTPHOST*** | ***200.10.194.33*** |
| • ... etc | |

Cette rubriaue existe encore :

/etc/hosts (UNIX)

c:\windows\hosts

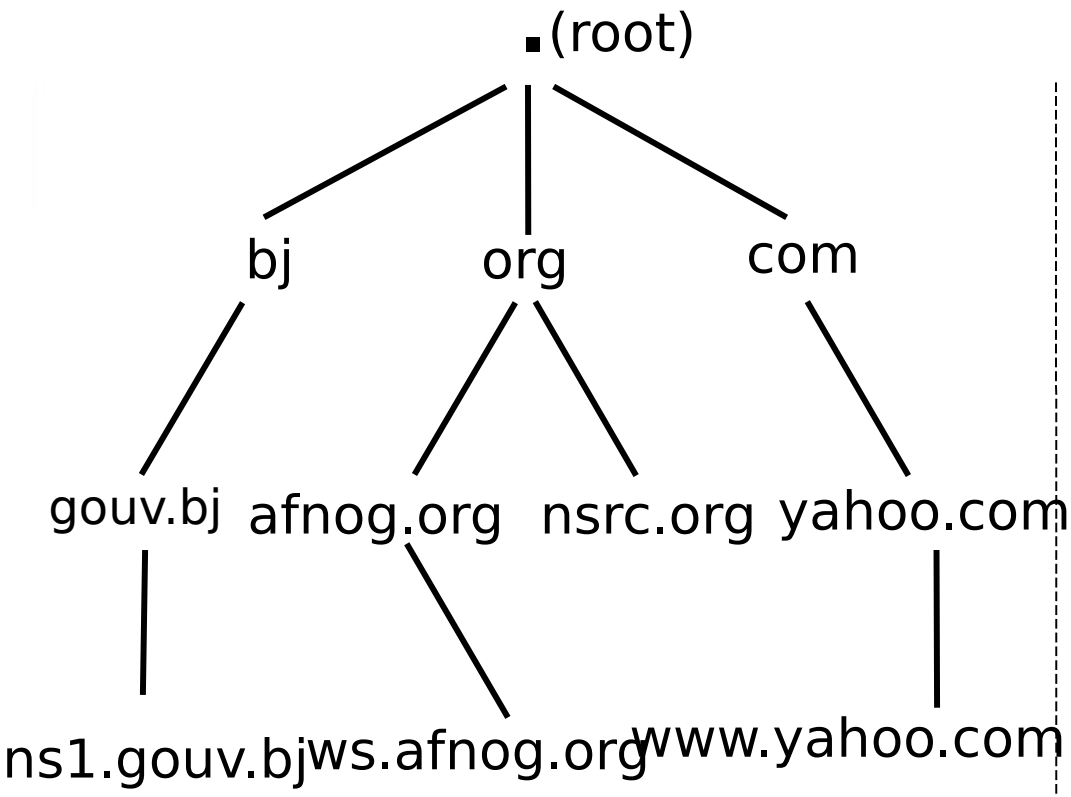
# Hosts.txt est inadapté à grande échelle

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

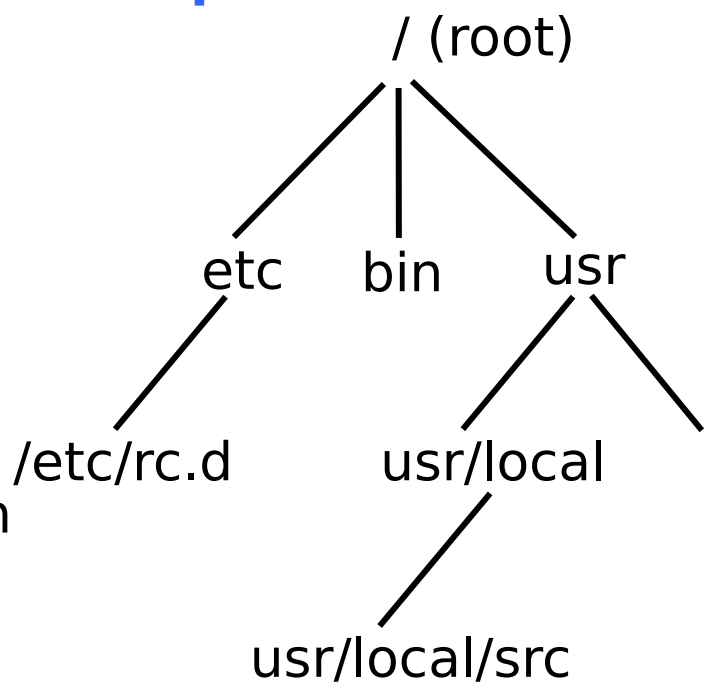
# Le Domain Name System est née

- 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)




Base de donnée DNS




Système de fichier Unix

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
- Example:
  - .net est sur un ensemble de serveurs de noms
  - ripe.net est sur un ensemble différent
  - disi.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) ou resource records RR

- 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 RRs

- A (adresse): associe le nom d'hôte à l'adresse IPv4
- AAAA (quad A): associe le nom d'hôte à l'adresse IPv6
- 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

Requête : www.afnog.org.
Requête de type: A
Résultat:

```
www.afnog.org.   14400   IN   A    196.216.2.4
```

**Dans ce cas un seul RR a été trouvé, ,mais en général, plusieurs RRs peuvent être retournés.**

(IN est la "classe" pour INTERNET utilisée 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 )
- Refusé (n'est pas autorisé à demander au serveur )

# 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 193.194.185.25**
***Nom de domaine: 25.185.194.193.in-addr.arpa.***
***Requête de Type: PTR***
***Resultat: ashanti.gh.com.***

***Connue comme"consulation inverse du DNS"*** (Parce que nous cherchons le nom pour une adresse IP, plutôt que l'adresse IP pour le nom)

# Questions?

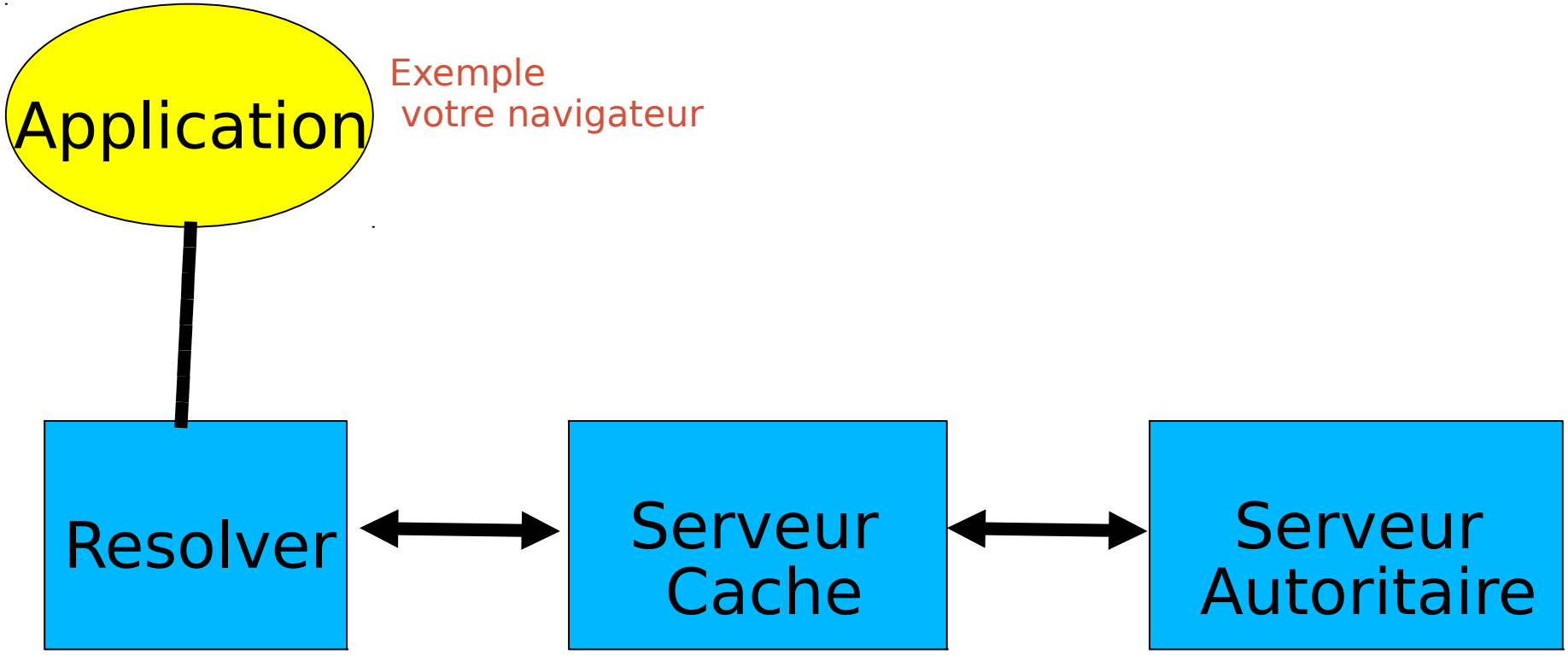
# Le DNS est une application Client/Serveur

- 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, ou un enregistrement IPv6 AAAA

# Trois rôles du 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  cache et du cache  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
- Ce qui est commun 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 hôte a besoin d'un resolver – ex chaque poste de travail Windows en a un

# Comment le resolver trouve 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 configurer ?

- Vous devez avoir la PERMISSION d'utiliser le serveur cache
  - Ex. serveur cache de votre ISP, ou le votre
- Préférer le serveur cache voisin
  - 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
- Éviter Habituellement

# Exemple: Configuration resolver Unix

/etc/resolv.conf

```
search ssf.ws.afnog.org
nameserver 196.200.219.200
nameserver 196.200.223.1
```

***C'est tout ceux 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 ce n’est pas un bon essai?

# Tester le DNS avec dig

- "dig" est un programme qui effectue des requêtes DNS et affiche les résultats
- Meilleur 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 particulier (dépasse
/etc/resolv.conf)
```

# Le point à la fin d'un nom de domaine

***dig tiscali.co.uk.***  *Trailing Dot*

- 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 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.t1.ws.afnog.org to SERVER: 81.199.110.100
;; WHEN: Sun Jun  8 21:18:18 2013
;; 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
  - REFUSED: la requête n'est pas disponible sur le serveur cache
- FLAGS
  - AA: Réponse des 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 la 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
  - vérifie quel serveur a donné la réponse!
  - Si vous faites une faute de frappe, la requête

# 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