

---

# DNS INVERSE ET GESTION DE LA DELEGATION INVERSE

Formation LIR/AFRINIC  
Ouaga du 10 au 11 Novembre 2004

Alain Patrick AINA  
aalain@trstech.net

# Lectures

---

- <http://www.rfc-editor.org/rfc/rfc1034.txt>
- <http://www.rfc-editor.org/rfc/rfc1035.txt>
- <http://www.rfc-editor.org/rfc/rfc2181.txt>
- <http://www.rfc-editor.org/rfc/rfc2182.txt>
- <http://www.rfc-editor.org/rfc/rfc1982.txt>
- <http://www.rfc-editor.org/rfc/rfc2317.txt>

# Adresses dans le DNS

---

- Faire correspondre les nombres aux noms
- C'est seulement un DNS ordinaire
  - ◆ Pas de normes différentes
  - ◆ Pas d'opérations différentes
- Mais vous aurez besoin de quelques informations
  - ◆ Il y a des conventions
  - ◆ IPv6 est un objectif encore en développement
- Ipv4 d'abord

# Correspondance des adresses à l'inverse

---

- La correspondance noms en adresses est courante:  
`www.afrinic.net`    A    `196.216.2.1`
- Parfois, on veut savoir le nom qui est lié a une adresse donnée. Si tu peux traduire l'adresse en nom complètement qualifié, on peut utiliser le DNS
- **But de la conception : Déléguer la maintenance du DNS inverse au propriétaire du bloc d'adresses.**

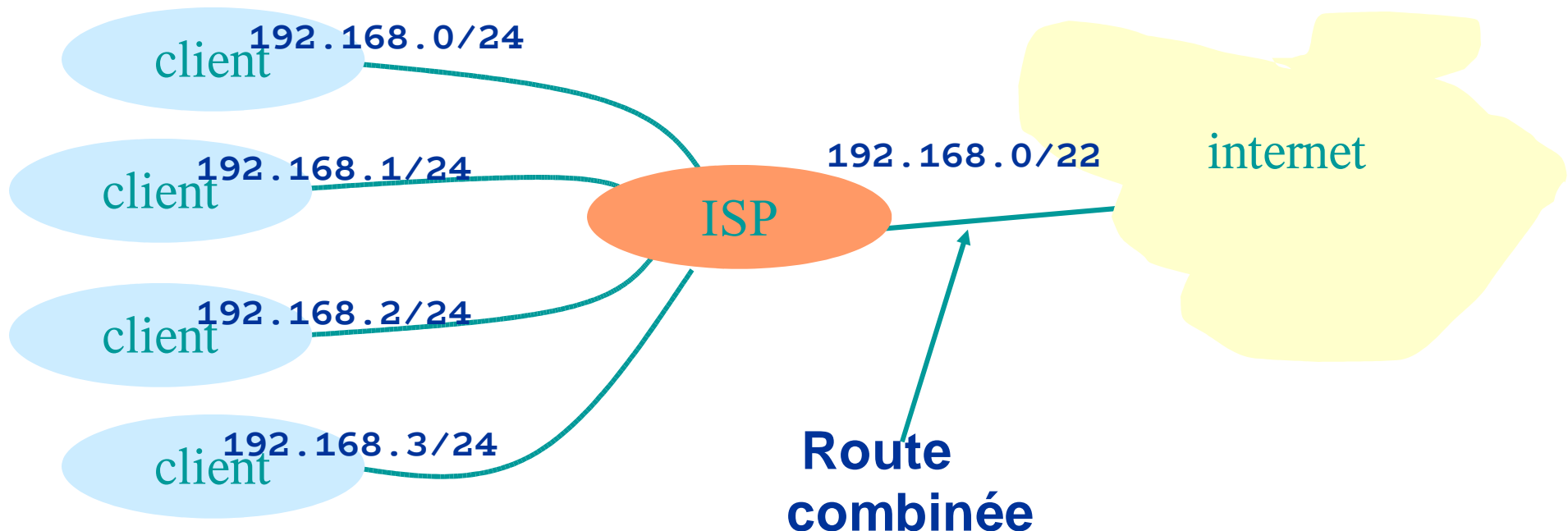
# Correspondances des adresses Ipv4 dans le DNS: attribution des adresses

---

- L'allocation des adresses est hiérarchique
  - ◆ Les blocs d'adresses sont alloués aux LIRs/ISPs
  - ◆ Des blocs d'adresses plus petits sont alloués aux clients
  - ◆ Les clients assignent des blocs d'adresses aux utilisateurs
- Le routage est basé sur la destination pour des blocs d'adresses donnés.
  - ◆ Historiquement sur 8 bit (Classe A,B,C)
  - ◆ Classless Inter Domain Routing (CIDR)

# Classless inter domain routing (CIDR)

- La taille de la table de routage (mémoire, processeur... de routeur) est une ressource limitée
- But de CIDR: rassembler beaucoup de petits blocs d'adresses en un bloc d'adresse plus grand



# Correspondances des adresses Ipv4 dans le DNS: bloc d'adresses

- Notation du bloc d'adresses :

<adresse>/<nombre de bits significatifs>

Par exemple :

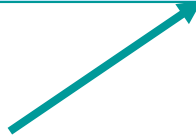
193.0.0.0/8 ou plus court 193/8

193.165.64/19=

0xc1a54000/19 =

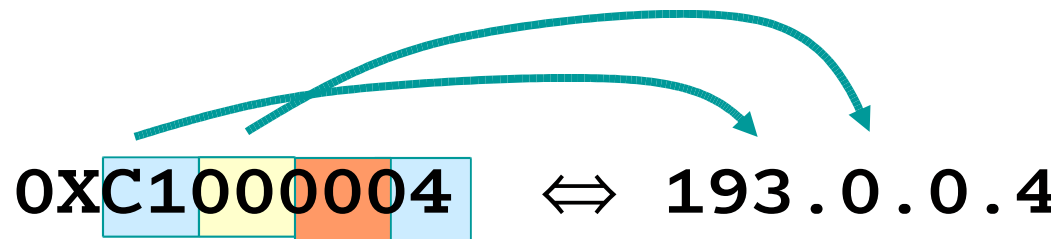
1100 0001 1001 0101 0010 0000 0000 0000

19  
bits



# Le format d'adresse IPv4

- Une adresse IP est un nombre normalement à 4 octets par une représentation décimale des 4 octets séparés par des points.



- L'allocation basée sur la limite de 8 bits, conduit à un simple schéma de délégation

# Correspondances des adresses Ipv4 dans le DNS

---

- Exemple 192.26.1.3
  - ◆ 192/8 est alloué au RIR
  - ◆ 192.26/16 est alloué par le RIR à un LIR/ISP
  - ◆ 192.26.1/24 est assigné par l'ISP à une entreprise.
- Délégation dans le DNS:
  - ◆ La racine délègue le domaine "192" au RIR
  - ◆ Le RIR délègue la sous-zone "26" à l'ISP
  - ◆ L'ISP délègue la sous-zone "1" à l'entreprise.
- Le nom qui rend ceci possible : 1.26.192

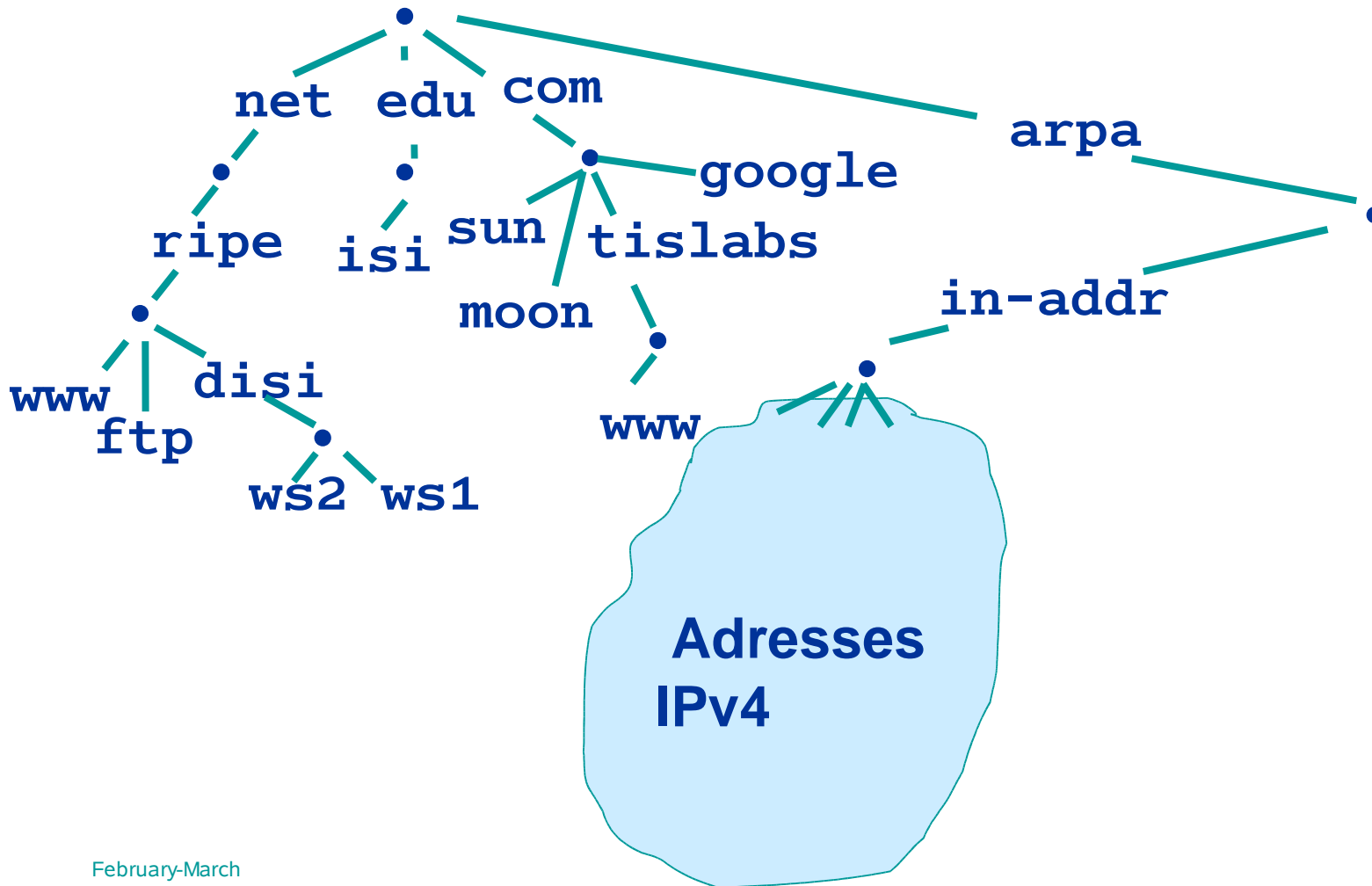
# Correspondances des adresses Ipv4 dans le DNS

---

- Inverser la représentation décimale :
  - ◆ 192.26.1.3 en 3.1.26.192 et mettre ceci sous un domaine de niveau supérieur
  - ◆ Pour IPv4 le domaine de niveau supérieur est **in-addr.arpa**
- Dans le DNS, on publie des enregistrements PTR qui pointent de nouveau vers le nom :

```
1.2.216.196.in-addr.arpa 3600 IN PTR www.afrinic.net.
```

# L'arbre inverse



# Correspondances des adresses Ipv4 dans le DNS

---

- En IPv4 la correspondance est faite à la limite de 8 bits (class full), l'allocation des adresses est class less
- L'administration de la zone ne chevauche pas toujours avec l'administration des adresses
- Si vous avez un /19 : diviser le en des /24 et demander la délégation pour chacun d'eux dès que vous utilisez le bloc d'adresses
- Nous couvrons les /25 et moins plus loin

# Configuration de zone inverse

---

- Le fichier de zone inverse est un fichier de zone normal.
  - ◆ Enregistrements SOA et NS dans l' APEX
  - ◆ Les enregistrements PTR dans la zone même
- Assurer vous que la zone est servie par plusieurs serveurs(maîtres et esclaves)
- Bind9 à la directive \$GENERATE qui pourrait être maniable

# Exemple de zone inverse

```
$ORIGIN 2.216.196.in-addr.arpa.  
$TTL 1d  
@ 3600 IN SOA ns.afrinic.net. hostmaster.afrinic.net. (  
    2004111100 ; serial  
    1h        ; refresh  
    30M       ; retry  
    1W        ; expiry  
    3600     ) ; neg. answ. ttl
```

```
NS ns.afrinic.net.  
NS ns.ripe.net.
```

```
1 PTR gw.afrinic.net.  
   router.afrinic.net.
```

```
2 PTR ns.afrinic.net.
```

```
; BIND9 va generer: 65 PTR machine65.afrinic.net.
```

```
$GENERATE 65-127 $ PTR machine$.afrinic.net.
```

Noter le point a la fin



# Obtenir une délégation inverse

---

- La procédure dépend du RIR
  - ◆ Pour la région RIPE, lire :  
[http://www.ripe.net/reverse/reverse\\_howto.html](http://www.ripe.net/reverse/reverse_howto.html)
- Seulement des délégations /16 et /24

# Données whois d'un objet domaine

---

```
domain:          186.56.62.in-addr.arpa
descr:          Netcom Togo, Togo
admin-c:        MH160-RIPE
tech-c:         AP147-RIPE
zone-c:         AP147-RIPE
nserver:        mac1.netcom.tg
nserver:        ns1.ipplanet.com
mnt-by:         AS12491-MNT
changed:        lir@ipplanet.net 20040731
source:         RIPE
```

# Allocation inférieure à /24

---

- Imaginer un bloc d'adresse /25 délégué à une société par un ISP
- La société veut maintenir le DNS inverse pour les adresses IP qu'elle utilise
- Dans le DNS inverse, la délégation n'est pas possible
- Utiliser la technique 'classless inaddr' décrite dans le RFC 2317
- Basée sur l'utilisation des CNAME
  - ◆ CNAME fournit un mécanisme pour faire des alias de noms vers d'autres espaces de nommage

# RFC2317 (1)

- 192.0.2.0/25 a organisation A,
- 192.0.2.128/26 a organisation B et
- 192.0.2.192/26 a organisation C

```
$ORIGIN 2.0.192.in-addr.arpa.  
;  
1 PTR machine1.organisationA.com.  
2 PTR machine2.organisationA.com.  
3 PTR machine3.organisationA.com.  
;  
129 PTR machine1.organisationB.com.  
130 PTR machine2.organisationB.com.  
131 PTR machine3.organisationB.com.  
;  
193 PTR machine1.organisationC.com.  
194 PTR machine2.organisationC.com.  
195 PTR machine3.organisationC.com.
```

# RFC2317 (2)

---

- Générer un sous-domaine pour chaque bloc d'adresse et déléguer le aux enfants
  - ◆ Nommer le sous-domaine après le bloc
    - ◆ 0/25, 128/26, et 190/26
    - ◆ 0-127, 128-189, 190-255
    - ◆ orgA, orgB, orgC
- Pour chaque nom dans la zone, créer un CNAME qui pointe vers l'espace de nommage délégué i.e.:

**1 CNAME 1.orgA.2.0.193.in-addr.arpa.**

# RFC2317 (3) zone du parent

---

```
$ORIGIN 2.0.192.in-addr.arpa.  
@      IN      SOA      mon-ns.mon.domain. hostmaster.mon.domain. (  
      ...)  
  
; ...  
orgA   NS ns1.organisationA.com.  
       NS ns2.organisationA.com.  
1      CNAME  1.orgA  
2      CNAME  2.orgA  
  
; ...  
orgB   NS ns1.organisationB.com.  
       NS ns2.organisationB.com.  
129    CNAME  129.orgB  
130    CNAME  130.orgB  
  
;
```

# RFC2317 (4) zone de l'enfant

---

```
$ORIGIN orgA.2.0.192.in-addr.arpa.  
@ IN SOA ns1.organisationA.com. hostmaster.organizationA.com(  
    ...)  
;  
@ NS ns1.organsationA.com.  
NS ns2.organisationA.com.  
1 PTR machine1.organisationA.com.  
2 PTR machine2.organisationA.com.
```

# RFC2317 (5)

- Vous pouvez aussi déléguer vers une zone de transfert
  - ◆ Facilite la maintenance

```
$ORIGIN 1.168.192.in-addr.arpa
;
;
24      CNAME      in24.foo.net.
25      CNAME      in25.foo.net.
26      CNAME      in26.foo.net.
27      CNAME      in27.foo.net.
28      CNAME      in28.foo.net.
;
; etc
```

```
$ORIGIN foo.net.
;
www A   192.168.1.24
in24 PTR www.foo.net.
ftp A   192.168.1.25
in25 PTR ftp.foo.net.
silver A 192.168.1.26
in26 PTR silver.foo.net.
;
; etc
```

