

Solution transitoire de mise à jour dynamique des adresses IPv4 et IPv6 dans le DNS

Nicolas Cuissard
Université Paris-1 Panthéon–Sorbonne

- Gestion du DNS
- Rappel EUI-64 / DDNS
- DHCPv4 + DHCPv6
- DDNS et adresses EUI-64
- Retour d'expérience

Gestion du DNS

- 2005 : Mise en place d'IPv6
- Développement d'un script « majdns »
- Hôtes LAN en EUI-64, DHCP « statique »
- 2010 : Refonte des LAN en DDNS

- Gestion du DNS
- **Rappel EUI-64 / DDNS**
- DHCPv4 + DHCPv6
- DDNS et adresses EUI-64
- Retour d'expérience

Rappel : IPv6 EUI-64

- Adresse configurée automatiquement
- Dépendante de l'adresse MAC

Préfixe	Adresse MAC
2001:660:3333:1a4f::/64	31:ff:41:1b:3b:10
↓	↓
2001:660:3333:1a4f:33ff:41 <u>ff</u> : <u>fe</u> 1b:3b10	

Rappel : IPv6 EUI-64

- Windows XP

```
ipv6 -p gpu UseTemporaryAddresses no
```

- Windows Seven

```
netsh interface ipv6 set privacy state=disabled store=active  
netsh interface ipv6 set privacy state=disabled store=persistent  
netsh interface ipv6 set global randomizeidentifiers=disabled store=active  
netsh interface ipv6 set global randomizeidentifiers=disabled store=persistent
```

- Ubuntu (/etc/sysctl.d/10-ipv6-privacy.conf)

```
net.ipv6.conf.all.use_tempaddr = 0  
net.ipv6.conf.default.use_tempaddr = 0
```

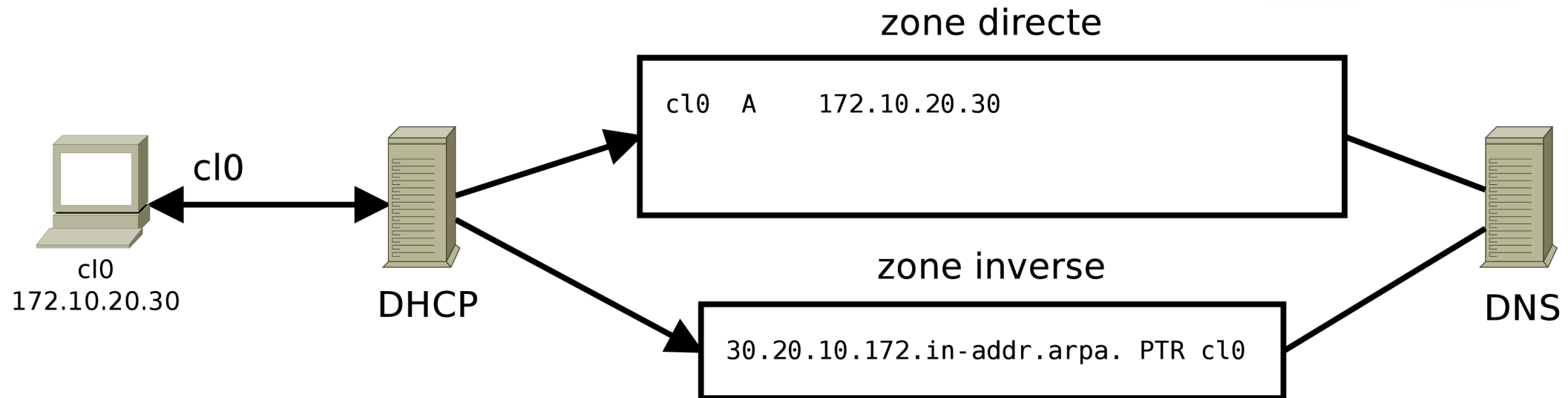
- Network Manager (/etc/NetworkManager/system-connections/*)

```
[ipv6]  
ip6-privacy=0
```

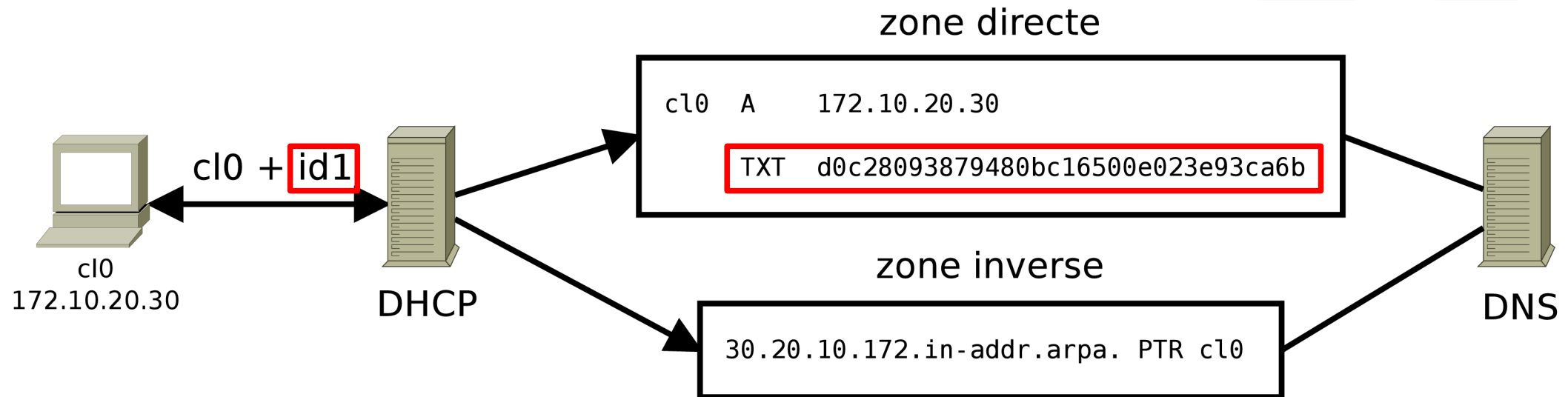
- OS X (/etc/sysctl.conf)

```
net.inet6.ip6.use_tempaddr=0
```

Rappel DDNS : principe RFC 4702 (v4), 4703 (v6)

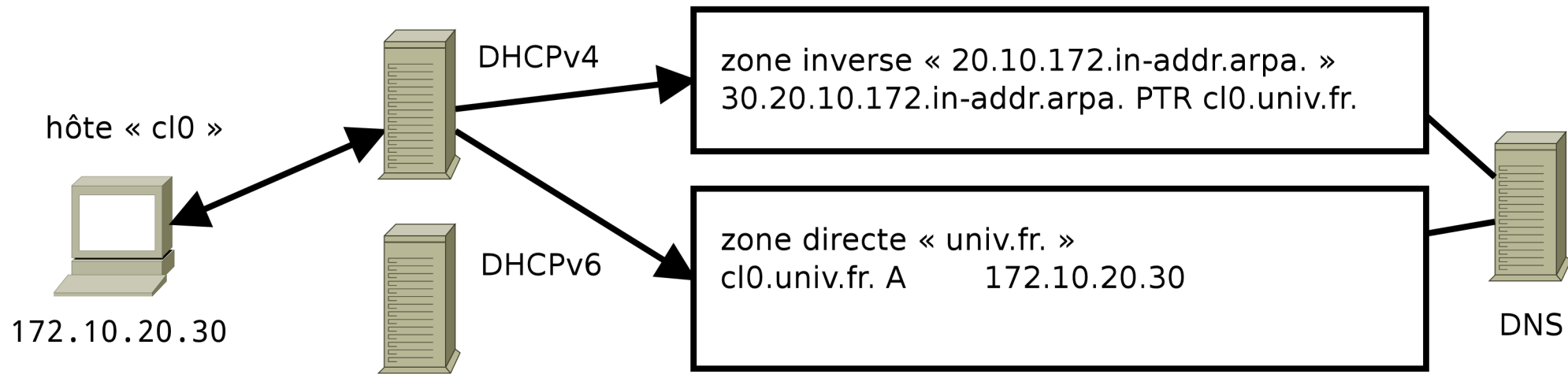


Rappel DDNS : détection des conflits RFC 4701

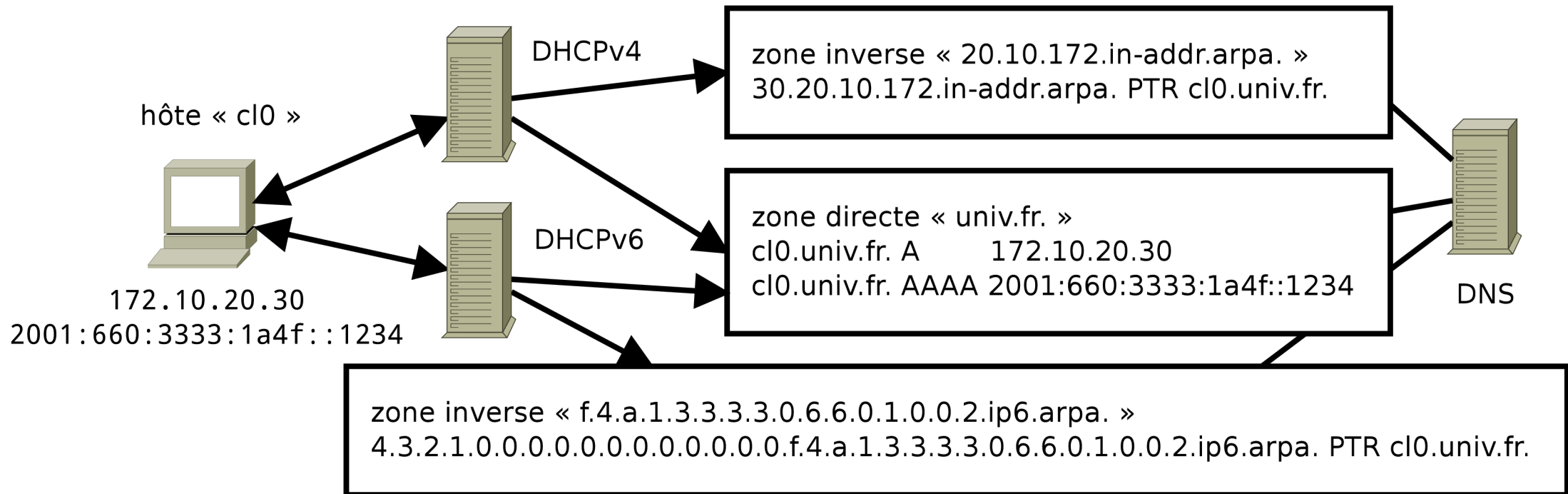


- Gestion du DNS
- Rappel EUI-64 / DDNS
- **DHCPv4 + DHCPv6**
- DDNS et adresses EUI-64
- Retour d'expérience

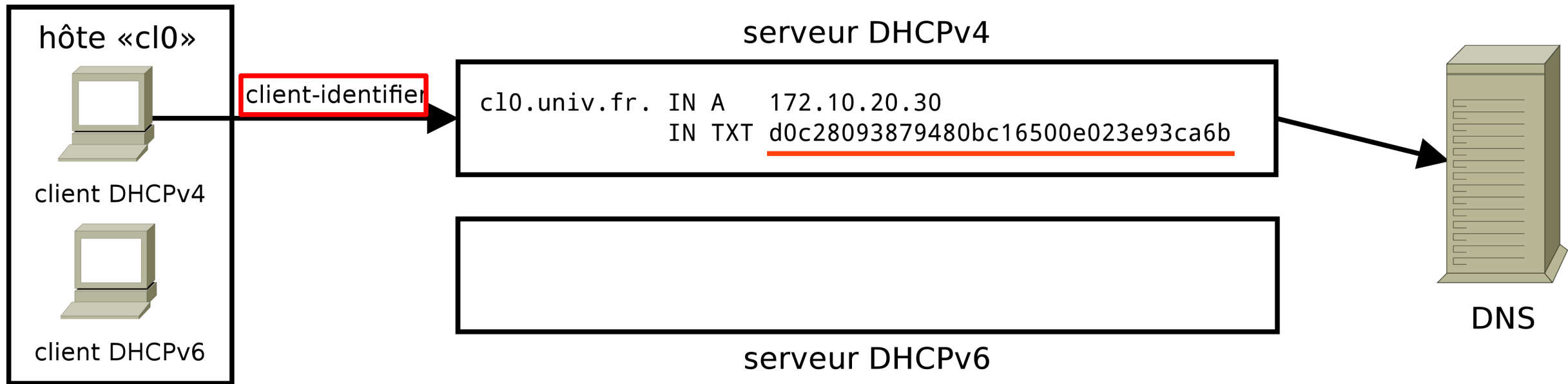
DHCPv4 + DHCPv6 : architecture



DHCPv4 + DHCPv6 : architecture

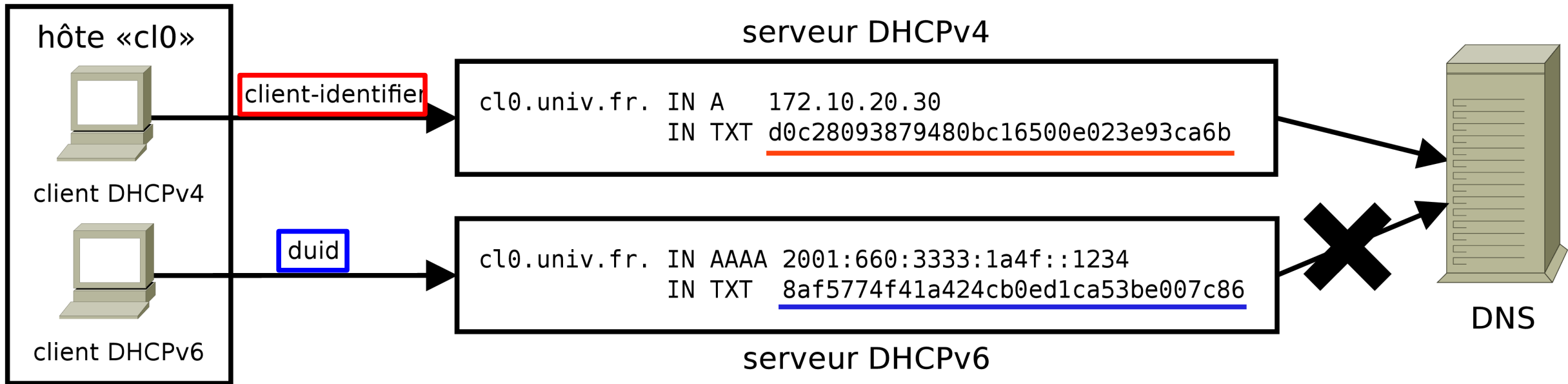


DHCPv4 + DHCPv6 : conflits



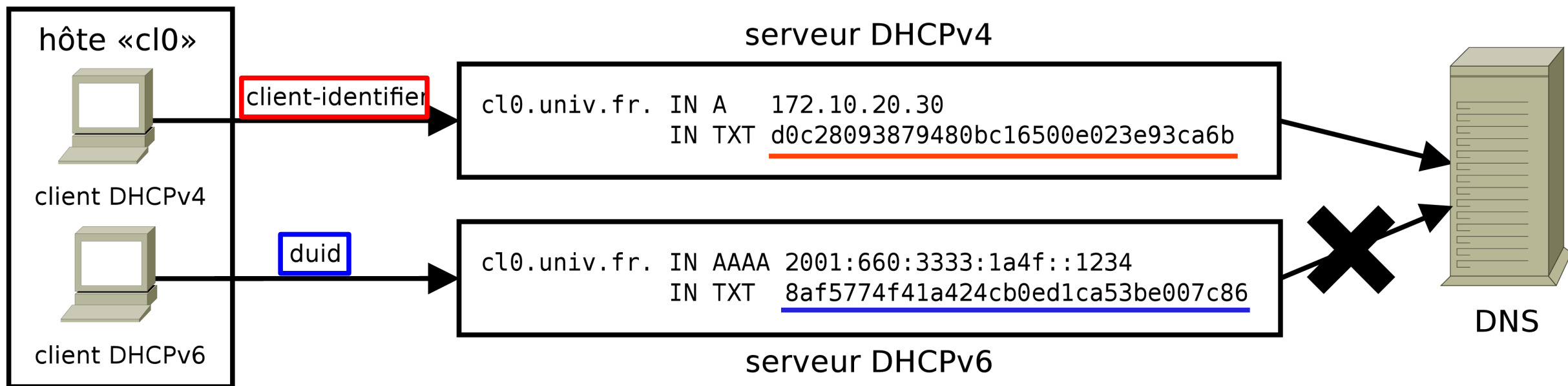
DHCPv4 + DHCPv6 : conflits

- Identifiant spécifique au protocole
(*client-identif* vs. DUID)



DHCPv4 + DHCPv6 : conflits

- Identifiant spécifique au protocole (*client-identif* vs. DUID)
- Pas de moyen simple pour faire le lien entre les identifiants / adresses

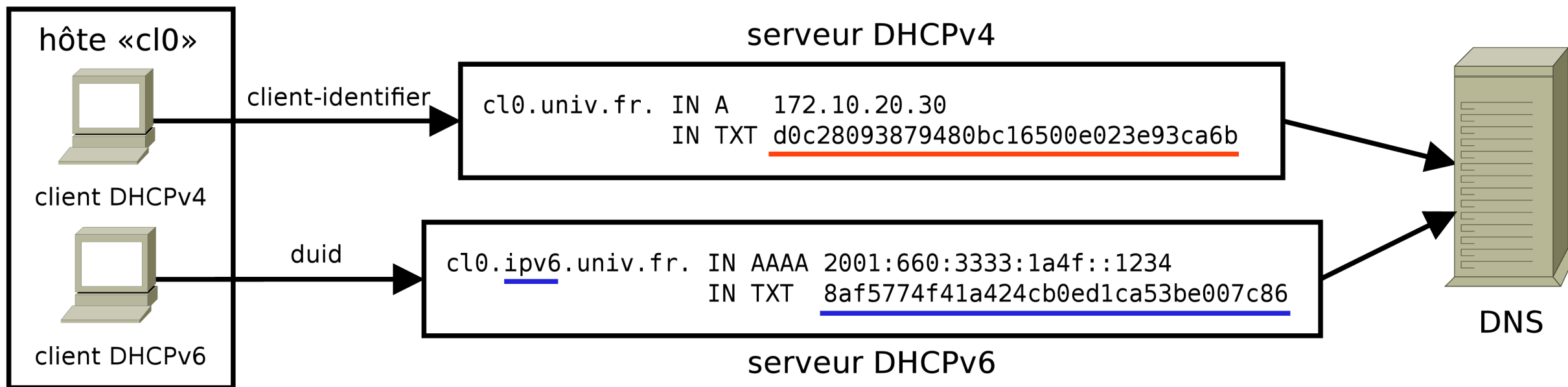


Conflits de mise à jour : solutions « radicales »

- Arrêter d'utiliser IPv4
- Arrêter d'utiliser IPv6
- Les hôtes effectuent le DDNS

DHCPv4 + DHCPv6 : deux zones directes

- Évite les conflits de mise à jour
- Ne garantit pas que les enregistrements proviennent du même hôte



Conflits DHCP : désactiver la détection des conflits sur DHCPv6

- Fonctionne en apparence
- Enregistrements IPv6 « open bar »
- Pollution des zones

1

démarrage du
client DHCPv4

```
cl0.univ.fr. IN A ● 172.10.20.30  
                IN TXT ● d0c28093879480bc16500e023e93ca6b
```

1

démarrage du
client DHCPv4

```
cl0.univ.fr. IN A ● 172.10.20.30  
                IN TXT ● d0c28093879480bc16500e023e93ca6b
```

2

démarrage du
client DHCPv6

```
cl0.univ.fr. IN A ● 172.10.20.30  
                IN AAAA ● 2001:660:3333:1a4f::1234  
                IN TXT ● 8af5774f41a424cb0ed1ca53be007c86
```

1

démarrage du
client DHCPv4

```
cl0.univ.fr. IN A ● 172.10.20.30  
                IN TXT ● d0c28093879480bc16500e023e93ca6b
```

2

démarrage du
client DHCPv6

```
cl0.univ.fr. IN A ● 172.10.20.30  
                IN AAAA ● 2001:660:3333:1a4f::1234  
                IN TXT ● 8af5774f41a424cb0ed1ca53be007c86
```

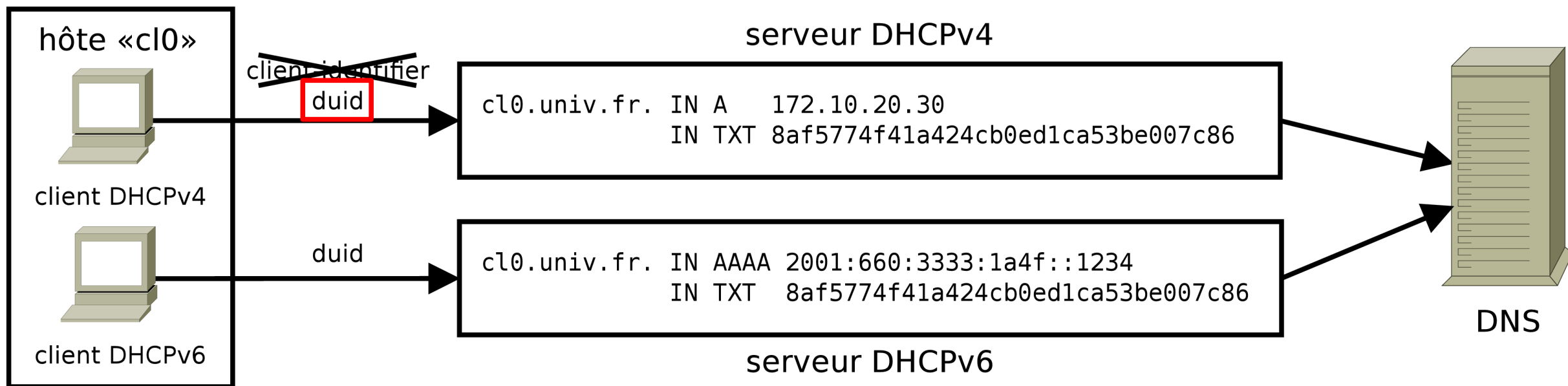
3

suppression de l'adresse IPv6
par libération ou expiration du bail

```
cl0.univ.fr. IN A ● 172.10.20.30  
                IN TXT ● 8af5774f41a424cb0ed1ca53be007c86
```

DHCPv4 + DHCPv6 : RFC 4361

- Les clients DHCP utilisent le même identifiant
- Même identifiant → même signature → pas de conflit



Conflits DHCP : RFC 4361 (2006)

- Solution la plus simple
- Permet d'avoir des serveurs indépendants
- ISC-DHCP 4.3 « *will allow v4 and v6 clients to cooperate on naming* »

Conflits DHCP : RFC 6939 (2013)

- Nouvelle option pour DHCPv6, adresse matérielle envoyée au serveur
- Le serveur doit faire le lien avec un bail DHCPv4 (?)
- Nécessite une communication entre serveurs ou un serveur unifié

Conflits DHCP : solutions spécifiques

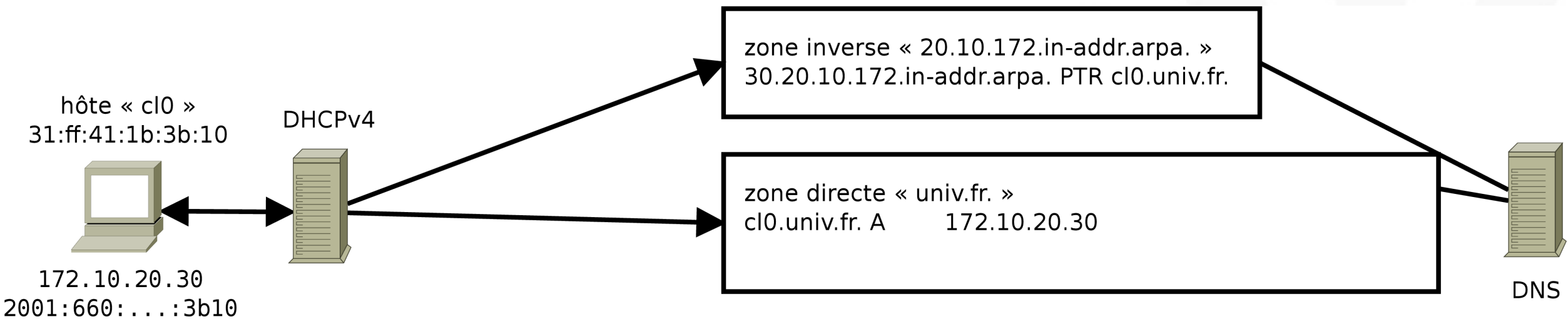
- Forcer un hôte IPv6 à passer par un service IPv4
- Utiliser des fonctionnalités de certains équipements actifs
- Clients de type *DynDNS*

- Gestion du DNS
- Rappel EUI-64 / DDNS
- DHCPv4 + DHCPv6
- **DDNS et adresses EUI-64**
- Retour d'expérience

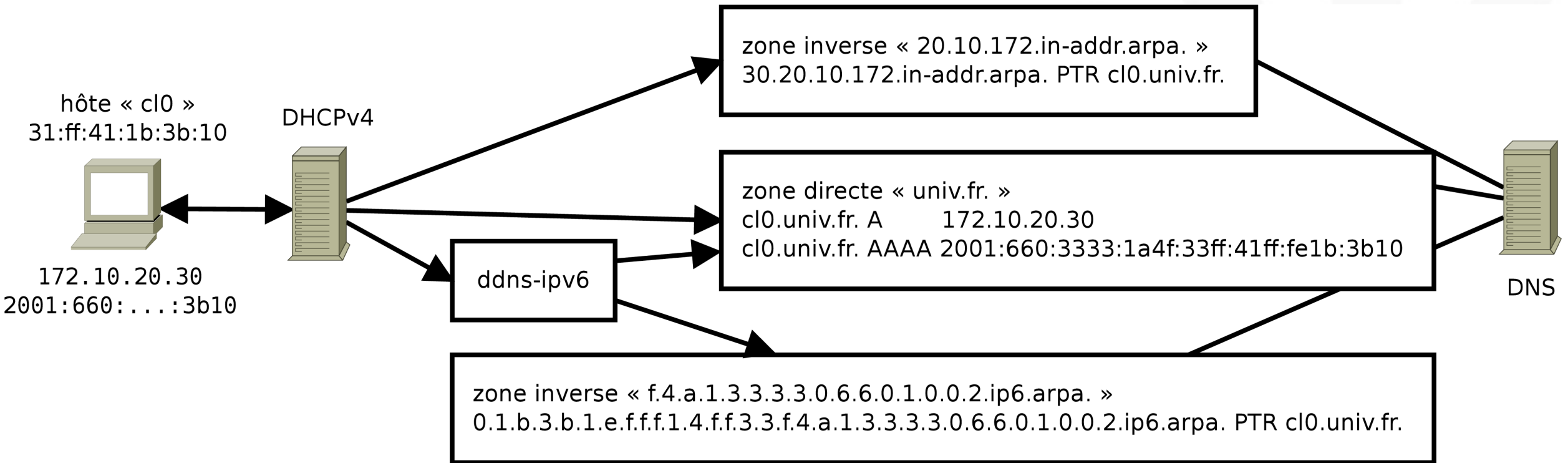
Solution temporaire : script « ddns-ipv6 »

- Université de Nouvelle-Galles du Sud
- Uniquement DHCPv4 : dhcp-eval + « execute »
- Mise à jour des IPv6 EUI-64 par script, validation par ICMPv6

Script « ddns-ipv6 »



Script « ddns-ipv6 »



Script « ddns-ipv6 » : limitations

- Pas de suppression des IPv6 à l'expiration du bail
→ mise en cache adresse mac et de l'IPv4
- Pas de détection des conflits de mise à jour
→ implémentation dans le script (requêtes DNS)
- Script exécuté systématiquement quelque soit le résultat du DDNS IPv4

Script « ddns-ipv6 » : configuration DHCP

```
set prefix = "2001:660:3333:1a4f";

on commit {
  if (not static) {
    set cache-ipv4 = binary-to-ascii(10, 8, ".", leased-address);

    set cache-mac = binary-to-ascii(16, 8, ":", substring(hardware, 1, 6));

    set ddns-hostname = option host-name;

  }
}
```

Script « ddns-ipv6 » : configuration DHCP

```
set prefix = "2001:660:3333:1a4f";
```

```
on commit {
```

```
  if (not static) {
```

```
    set cache-ipv4 = binary-to-ascii(10, 8, ".", leased-address);
```

```
    set cache-mac = binary-to-ascii(16, 8, ":", substring(hardware, 1, 6));
```

```
    set ddns-hostname = option host-name;
```

```
    execute ("/usr/local/bin/ddns-ipv6", prefix,  
            ddns-hostname, cache-mac, cache-ipv4);
```

```
  }
```

```
}
```

} ajout

Script « ddns-ipv6 » : configuration DHCP

```
set prefix = "2001:660:3333:1a4f";
```

```
on commit {
```

```
  if (not static) {
```

```
    set cache-ipv4 = binary-to-ascii(10, 8, ".", leased-address);
```

```
    set cache-mac = binary-to-ascii(16, 8, ":", substring(hardware, 1, 6));
```

```
    set ddns-hostname = option host-name;
```

```
    execute ("/usr/local/bin/ddns-ipv6", prefix,  
            ddns-hostname, cache-mac, cache-ipv4);
```

```
  }
```

```
}
```

```
on release or expiry {
```

```
  execute ("/usr/local/bin/ddns-ipv6", prefix,  
          "-d", ddns-hostname, cache-mac, cache-ipv4);
```

```
}
```

ajout

suppression

Script « ddns-ipv6 » : bilan

- DHCPv4 et EUI-64 : standard et indépendant de l'OS ou du matériel
- Script bash externe (dig, nsupdate, ipv6calc)
→ pas de patch d'ISC-DHCP
- Fonctionne en mode *failover*

- Gestion du DNS
- Rappel EUI-64 / DDNS
- DHCPv4 + DHCPv6
- DDNS et adresses EUI-64
- **Retour d'expérience**

Migration des hôtes

- Environ 27% des hôtes font le DDNS IPv6
- Problèmes de DDNS indépendants du script
- Dans la continuité de notre utilisation d'IPv6

Problème DDNS DHCPv4

- A ou TXT orphelins, PTR manquants
- Améliorations avec ISC-DHCP 4.2.x
- Fiabilisation avec le script « ddns-ipv6 »
- Vérification occasionnelle des zones par script

Évolution de l'infrastructure

- Utiliser le *client-identifier* pour la détection des conflits de mise à jour
- Utiliser DHCPv6 en parallèle pour se préparer
- Clients compatibles RFC 4361 ?
Migrer de « ddns-ipv6 » → DHCPv6
- Adresses IPv6 dynamiques dans la zone externe ?

Conclusion

- Identifiants DHCPv4/v6 différents → conflits de DDNS
- Solution : RFC 4361 mais pas implémentée dans les OS
- Solution transitoire : ddns-ipv6 / EUI-64 fonctionne

« So, welcome to the migration
period. »

David W. Hankins, 2010