

Le 7 octobre 2004

Exercice 1 : Création d'une table de routage

EXO: Ajouter une route à la table de routage d'un réseau [10.0.0.0](#) avec masque de réseau [255.0.0.0](#) et de passerelle [192.168.13.21](#)

AVANT:

```
linux:/home/aure # route
```

Table de routage IP du noyau

Destination	Passerelle	Genmask	Indic	Metric	Ref	Use	Iface
192.168.13.0	*	255.255.255.0	U	0	0	0	eth0
link-local	*	255.255.0.0	U	0	0	0	eth0
loopback	*	255.0.0.0	U	0	0	0	lo
default	baby.laboinfo	0.0.0.0	UG	0	0	0	eth0

COMMANDE:

```
linux:/home/aure # route add -net 10.0.0.0 netmask 255.0.0.0 gw 192.168.13.21
```

```
linux:/home/aure # route
```

APRES:

Table de routage IP du noyau

Destination	Passerelle	Genmask	Indic	Metric	Ref	Use	Iface
192.168.13.0	*	255.255.255.0	U	0	0	0	eth0
link-local	*	255.255.0.0	U	0	0	0	eth0
10.0.0.0	192.168.13.21	255.0.0.0	UG	0	0	0	eth0
loopback	*	255.0.0.0	U	0	0	0	lo
default	baby.laboinfo	0.0.0.0	UG	0	0	0	eth0

```
route add -net 10.0.0.0 netmask 255.0.0.0 gw 192.168.13.21
```

metric1 permet de donner une préférence --> le plus petit = le préféré.

dev eth0 permet de choisir la carte réseau s'il y a deux dans une machine.

Packet envoyé :

Ad. Source	Port source	Ad. destinataire	Port dest.	
192.168.13.114	8756	10.0.0.56	IMCP	Mode routage
10.0.0.1	7156			Mode NAT

Il y a une table de routage à l'emplacement

```
linux:/home/aure # cd /etc/sysconfig
```

```
linux:/etc/sysconfig # ls
```

```
.      cron      joystick nfs      sound
..     cups      kernel  onlineupdate  ssh
apache displaymanager keyboard personal-firewall suseconfig
autofs esound    language postfix    SuSEfirewall2
autoinstall fonts-config ldap    powersave    susehelp
backup hardware lirc    proxy        sysctl
boot   hotplug   mail    saslauthd    syslog
bootloader ide      mdadm   sax          windowmanager
bootsplash isdn    mouse   scripts      xntp
clock  ispell  network scsidev      ypbind
console java    news    security
```

```
linux:/etc/sysconfig # cd network
```

```
linux:/etc/sysconfig/network # ls
```

```
.    ifcfg-eth-id-00:0c:29:62:13:91 ifroute-lo scripts
..   ifcfg-lo                if-up.d    update-log
```

```
config ifcfg.template          providers update-log.debug
dhcp if-down.d                 routes wireless
linux:/etc/sysconfig/network # vi ifcfg.template
```

On recommande de créer un fichier par adresse IP différente sur la carte.

```
ex:  ifcfg-eth0:1
      ifcfg-eth0:2
```

Installation d'un serveur BIND9

Installer le paquetage nommé *BIND*.

ps -A : permet de lister les processus en cours d'exécution.

named n'est pas présent dans cette liste.

Allons voir dans le dossier */etc/init.d* et exécutons *./named*. Le serveur démarre.

Pour vérifier le bon déroulement de cette opération, il suffit d'aller voir les logs. Ils sont situés dans */var/log*. On va vérifier le contenu du fichier *messages*.

Exemple de fichier messages :

```
Oct 7 11:41:19 linux su: pam_unix2: session finished for user aure, service su
Oct 7 11:41:45 linux kernel: eth0: no IPv6 routers present
Oct 7 11:44:53 linux named[7367]: starting BIND 9.2.3 -t /var/lib/named -u named
Oct 7 11:44:53 linux named[7367]: using 1 CPU
Oct 7 11:44:53 linux named[7367]: loading configuration from '/etc/named.conf'
Oct 7 11:44:53 linux named[7367]: listening on IPv6 interfaces, port 53
Oct 7 11:44:53 linux named[7367]: listening on IPv4 interface lo, 127.0.0.1#53
Oct 7 11:44:53 linux named[7367]: binding TCP socket: address in use
Oct 7 11:44:53 linux named[7367]: listening on IPv4 interface eth0, 192.168.13.214#53
Oct 7 11:44:53 linux named[7367]: binding TCP socket: address in use
Oct 7 11:44:53 linux named[7367]: command channel listening on 127.0.0.1#953
Oct 7 11:44:53 linux named[7367]: command channel listening on ::1#953
Oct 7 11:44:53 linux named[7367]: zone 0.0.127.in-addr.arpa/IN: loaded serial 42
Oct 7 11:44:53 linux named[7367]: zone localhost/IN: loaded serial 42
Oct 7 11:44:53 linux named[7367]: runningOct 7 11:41:19 linux su: pam_unix2: session finished for user aure, service
su
Oct 7 11:41:45 linux kernel: eth0: no IPv6 routers present
Oct 7 11:44:53 linux named[7367]: starting BIND 9.2.3 -t /var/lib/named -u named
Oct 7 11:44:53 linux named[7367]: using 1 CPU
Oct 7 11:44:53 linux named[7367]: loading configuration from '/etc/named.conf'
Oct 7 11:44:53 linux named[7367]: listening on IPv6 interfaces, port 53
Oct 7 11:44:53 linux named[7367]: listening on IPv4 interface lo, 127.0.0.1#53
Oct 7 11:44:53 linux named[7367]: binding TCP socket: address in use
Oct 7 11:44:53 linux named[7367]: listening on IPv4 interface eth0, 192.168.13.214#53
Oct 7 11:44:53 linux named[7367]: binding TCP socket: address in use
Oct 7 11:44:53 linux named[7367]: command channel listening on 127.0.0.1#953
Oct 7 11:44:53 linux named[7367]: command channel listening on ::1#953
Oct 7 11:44:53 linux named[7367]: zone 0.0.127.in-addr.arpa/IN: loaded serial 42
Oct 7 11:44:53 linux named[7367]: zone localhost/IN: loaded serial 42
Oct 7 11:44:53 linux named[7367]: running
```

Maintenant, le fichier *named.conf*

Différents types :

master : serveur DNS principal

slave : serveur DNS de sauvegarde

hint : serveurs root

Il faut ajouter sa propre zone :

Ex :

```
zone « baltazar » in {  
    type master;  
    file « baltazar.zone »;  
};
```

Créer le fichier de zone correspondant dans le dossier *master* ou *slave* selon le type de serveur à mettre en place.

Redémarrer le serveur DNS par la commande */etc/init.d/named restart*

Au niveau du fichier *resolv.conf*:

rajouter *search baltazar* pour que le serveur sache qu'il faut aussi chercher dans la zone *balthazar*.
On pourrait aussi placer *helho.be*. En pingant *www*, on verrait qu'on obtient automatiquement www.helho.be. Attention, placer toutes les zones sur la même ligne de *search*.

Remarques :

- ne pas oublier de changer la configuration réseau de chaque pc :
Nom du domaine
Adresse du serveur de noms, domaine sur lequel il a autorité
- sur le serveur : ne pas oublier de laisser le serveur DNS principal (dans le cas de l'école).

Si on crée un serveur esclave : modifier le fichier *named.conf* et ajouter la série de lignes analogues suivantes :

```
zone « baltazar » in {  
    type slave;  
    file « ./slave/baltazar.zone »;  
    masters { 192.168.13.142 };  
};
```

On peut vérifier dans le fichier de log */var/log/messages* que le transfert de zone a bien été effectué.
Pour filtrer les demandes de transferts de zones, on modifie la ligne *allow-query* du fichier *named.conf*.

Pour notifier à l'esclave qu'il y a eu modification dans une zone dont il est responsable :

- on modifie le paramètre de la ligne *notify* : *notify explicit*
ajouter une ligne *also_notify* : {192.168.13.x} ; adresse IP de l'esclave