

# Configuration réseau Basique

## 1. Configuration réseau bas niveau

Les outils de configuration réseau bas niveau traditionnels des systèmes GNU/Linux sont les programmes `ifconfig` et `route` qui viennent dans le paquet `net-tools`. Ces outils ont officiellement été rendus obsolètes par `ip` qui est dans le paquet `iproute`. Le programme `ip` fonctionne avec Linux 2.2 et supérieurs et a plus de possibilités que les anciens outils. Cependant, les anciens outils fonctionnent toujours et sont plus familiers à de nombreux utilisateurs.

### a. Configuration réseau bas niveau – `ifconfig` et `route`

Voici une illustration de comment changer l'adresse IP de l'interface `eth0` de `192.168.0.3` en `192.168.0.111` et faire de `eth0` la route vers le réseau `10.0.0.0` via `192.168.0.1`. On commence par exécuter `ifconfig` et `route` sans arguments pour afficher l'état courant de toutes les interfaces réseau et du routage.

```
# ifconfig
eth0 Link encap:Ethernet HWaddr 08:00:46:7A:02:B0
     inet addr:192.168.0.3 Bcast:192.168.0.255 Mask:255.255.255.0
     UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
     RX packets:23363 errors:0 dropped:0 overruns:0 frame:0
     TX packets:21798 errors:0 dropped:0 overruns:0 carrier:0
     collisions:0 txqueuelen:100
     RX bytes:13479541 (12.8 MiB) TX bytes:20262643 (19.3 MiB)
     Interrupt:9

lo  Link encap:Local Loopback
     inet addr:127.0.0.1 Mask:255.0.0.0
     UP LOOPBACK RUNNING MTU:16436 Metric:1
     RX packets:230172 errors:0 dropped:0 overruns:0 frame:0
     TX packets:230172 errors:0 dropped:0 overruns:0 carrier:0
     collisions:0 txqueuelen:0
     RX bytes:22685256 (21.6 MiB) TX bytes:22685256 (21.6 MiB)

# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
192.168.0.0 * 255.255.0.0 U 0 0 0 eth0
default 192.168.0.1 255.255.255.255 UG 0 0 0 eth0
```

D'abord, on désactive l'interface.

```
# ifconfig eth0 inet down
# ifconfig
lo  Link encap:Local Loopback
    ... (plus d'entrée eth0)
# route
    ... (plus d'entrée dans la table de routage)
```

Ensuite, on l'active avec la nouvelle adresse IP et le nouveau routage.

```
# ifconfig eth0 inet up 192.168.0.111 \  
    netmask 255.255.0.0 broadcast 192.168.255.255  
# route add -net 10.0.0.0 netmask 255.0.0.0 gw 192.168.0.1 dev eth0
```

Le résultat :

```
# ifconfig  
eth0 Link encap:Ethernet HWaddr 08:00:46:7A:02:B0  
    inet addr:192.168.0.111 Bcast:192.168.255.255 Mask:255.255.0.0  
    UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
    ...  
  
lo  Link encap:Local Loopback  
    inet addr:127.0.0.1 Mask:255.0.0.0  
    ...  
# route  
Kernel IP routing table  
Destination Gateway Genmask Flags Metric Ref Use Iface  
192.168.0.0 * 255.255.0.0 U 0 0 0 eth0  
10.0.0.0 192.168.0.1 255.0.0.0 UG 0 0 0 eth0
```

Pour plus d'information voir `ifconfig(8)` et `route(8)`.

## b. Configurer une interface Wi-Fi

Pour les interfaces Wi-Fi, le programme `iwconfig` qui vient dans le paquet `wireless-tools` est utilisé en plus de `ifconfig` ou `ip`.

## 2. Nommer l'ordinateur

### a. Nom de machine

Un système Debian a parfois besoin de s'identifier par un nom. Pour cela, un **nom de machine** est maintenu par le noyau.

Le script d'initialisation `/etc/init.d/hostname.sh` définit le nom de machine au démarrage (en utilisant la commande `hostname`) au nom donné dans `/etc/hostname`. Ce fichier doit contenir **seulement** le nom de machine, pas un nom de domaine complet.

Pour afficher le nom de machine actuel, exécutez `hostname` sans argument.

## 3. Domain Name Service (DNS)

Les machines sont identifiées par un nom de domaine comme par une adresse IP. DNS est un système client-serveur dans lequel des résolveurs de noms consultent des serveurs de noms afin d'associer les noms de domaine avec les adresses IP et d'autres propriétés des machines. La bibliothèque GNU C `resolver(3)` peut aussi chercher des adresses IP dans des fichiers ou consulter Network Information Services (NIS).

Pour voir quel nom de domaine est associé avec la machine locale, utilisez la commande `hostname --fqdn`. Cela affiche le premier nom de domaine complet que le résolveur trouve pour la machine locale. [48]

### a. Le résolveur

Trouver quelle adresse IP est associée avec un domaine particulier est le travail du résolveur. Le résolveur le plus communément utilisé est l'ensemble de fonctions qui ont ce nom (`resolver(3)`) dans la bibliothèque GNU C. Un autre est le résolveur FireDNS qui vient dans le paquet `libfiredns`.

La façon dont le résolveur de la bibliothèque C résout les noms est définie par la ligne `hosts` du fichier de configuration `/etc/nsswitch.conf`. Cette ligne liste les services qui doivent être utilisés pour résoudre un nom : par exemple, `dns`, `files`, `nis`, `nisplus`. [49] Voir `nsswitch.conf(5)`. Si le service `files`, le comportement est aussi défini par le fichier de configuration `/etc/hosts`. Voir `hosts(5)`.

Tous les fichiers ci-dessus sont statiques et peuvent être édités avec votre éditeur favori.

Si le service `dns` est utilisé, le comportement du résolveur est aussi défini par le fichier de configuration `/etc/resolv.conf`. Voir `resolv.conf(5)`. Une des fonctions importantes de `/etc/resolv.conf` est de lister les adresses IP des serveurs de noms qui seront contactés pour résoudre le nom. Cette liste dépend souvent de l'environnement réseau, qui peut changer de temps en temps pendant que votre machine tourne. Des programmes comme `pppd` et `dhclient` sont capables de manipuler `resolv.conf` pour y ajouter ou enlever des lignes, mais ces possibilités ne marchent pas toujours correctement et peuvent créer des conflits mutuels. Le paquet `resolvconf` résout le problème en fournissant un cadre standard pour mettre à jour ce fichier.

### b. Gérer l'information sur les serveurs de noms – `resolvconf`

Le paquet `resolvconf` fournit un cadre pour la gestion dynamique des informations sur les serveurs de noms disponibles. Il résout le vieux problème du maintien de listes dynamiques de serveurs de noms pour le résolveur et le cache DNS. `resolvconf` se place comme intermédiaire entre les programmes qui contrôlent les interfaces réseau et fournissent l'information sur les serveurs de noms, et les applications qui ont besoin de l'information sur les serveurs de noms.

`resolvconf` est conçu pour fonctionner sans aucune configuration manuelle préalable. Cependant, le paquet est relativement nouveau et peut nécessiter des interventions manuelles pour qu'il fonctionne correctement. C'est certainement vrai si vous avez personnalisé des paquets pour qu'ils mettent à jour `/etc/resolv.conf` : il faudra désactiver ces modifications.

### c. Garder en mémoire les noms résolus – dnsmasq

Sauf si votre serveur de nom est le serveur officiel d'un domaine, il vaut mieux utiliser un serveur de noms cache local comme dnsmasq. Il fonctionne bien avec resolvconf.

## 4. Configuration réseau haut niveau avec Debian

Afin de permettre la configuration réseau plus facile Debian fournit un outil de configuration haut niveau standard qui comprend les commandes ifup et ifdown et le fichier `/etc/network/interfaces`. [\[51\]](#) Si vous choisissez d'utiliser ifupdown pour configurer votre réseau alors vous ne devriez pas utiliser les commandes bas niveau aussi. [\[52\]](#) Ifupdown est écrit avec l'hypothèse qu'il sera seul utilisé pour configurer et déconfigurer les interfaces réseau.

Pour mettre à jour la configuration des interfaces, faites ceci :

```
# ifdown eth0
# editor /etc/network/interfaces # modifiez à votre convenance
# ifup eth0
```

### a. Configurer une interface avec une adresse IP fixe

Supposons que vous voulez configurer une interface Ethernet de façon à avoir l'adresse IP fixe 192.168.0.123. Cette adresse commence par 192.168.0 donc elle doit être sur un réseau local. Supposons aussi que 192.168.0.1 est l'adresse de la passerelle du réseau local vers l'Internet. Éditez `/etc/network/interfaces` et ajoutez une partie comme suit :

```
iface eth0 inet static
    address 192.168.0.123
    netmask 255.255.255.0
    gateway 192.168.0.1
```

Si vous avez le paquet `resolvconf` installé alors vous pouvez ajouter ces lignes pour spécifier l'information DNS. Par exemple :

```
iface eth0 inet static
    address 192.168.0.123
    netmask 255.255.255.0
    gateway 192.168.0.1
    dns-search nicedomain.org
    dns-nameservers 195.238.2.21 195.238.2.22
```

Après que l'interface ait été activée, les arguments des options `dns-search` et `dns-nameservers` sont disponibles à `resolvconf` pour inclusion dans `resolv.conf`. L'argument `nicedomain.org` de l'option `dns-search` correspond à l'argument d'une option `search` de `resolv.conf(5)`. Les arguments `195.238.2.21` et `195.238.2.22` de l'option `dns-nameservers` correspondent aux arguments des options `nameserver` de `resolv.conf(5)`. D'autres options sont `dns-domain` et `dns-sortlist`.

## **b. Configurer une interface par DHCP**

Pour configurer une interface par DHCP, éditez le fichier `/etc/network/interfaces` afin d'y inclure :

```
iface eth0 inet dhcp
```

Afin que cela fonctionne vous devez avoir installé l'un des clients DHCP mentionnés dans

## **c. Configurer une interface Wi-Fi**

Le paquet `wireless-tools` inclut un script `/etc/network/if-pre-up.d/wireless-tools` qui permet de configurer du matériel Wi-Fi (802.11a/b/g) avant qu'une interface soit activée. La configuration se fait à l'aide du programme `iwconfig` ; voir `iwconfig(8)`. Pour chaque paramètre de la commande `iwconfig` vous pouvez inclure une option dans `/etc/network/interfaces` avec le nom du paramètre préfixé par « `wireless-` ». Par exemple, pour configurer l'ESSID de `eth0` en `myessid` et la clé de chiffrement à `123456789e` avant d'activer `eth0` par DHCP, éditez `/etc/network/interfaces` comme suit :

```
iface eth0 inet dhcp
    wireless-essid myessid
    wireless-key 123456789e
```

## **d. Configurer plusieurs interfaces Ethernet pour une passerelle**

Supposons que `eth0` soit connectée à l'Internet avec une adresse IP obtenue par DHCP et `eth1` soit connectée au LAN avec l'adresse IP fixe `192.168.1.1`. Éditez `/etc/network/interfaces` pour y inclure :

```
iface eth0 inet dhcp

iface eth1 inet static
    address 192.168.1.1
    netmask 255.255.255.0
```

Si vous activez NAT sur cette machine comme décrit dans alors vous pourrez partager la connexion Internet avec toutes les machines du LAN.