

# LA COMMANDE IPROUTE2

**Syntaxe :** ip [ OPTIONS ] OBJET { COMMANDE | help }

**Avec :** OBJET := { link | addr | route | rule | neigh | tunnel | maddr | mroute | monitor }  
OPTIONS := { -v[ersion] | -s[tatistics] | -r[esolve] | -f[amily] { inet | inet6 | ipx | dnet | link }  
| -o[neline] }

## 1. LA COMMANDE IP LINK

### 1.1 Ip link set

**Ip link set** modifie les attributs de l'interface. Abréviations : set, s

```
ip link set [ dev NAME ] [ {up | down} ] [ arp {on | off} ] [ name NAME ]  
[ txqlen NUMBER ] [ mtu NUMBER ] [ address LLADDRESS ]  
[ brd LLADDRESS ]
```

Il a pour paramètres :

- dev NAME : NAME indique l'interface réseau sur laquelle opérer ;
- up et down : change l'état de l'interface, respectivement actif et inactif ;
- arp on ou arp off : change le flag NOARP ;
- multicast on ou multicast off : change le flag MULTICAST ;
- name NAME : change le nom ;
- txqlen NUMBER ou txqlen NUMBER : change la taille de la file d'attente de l'interface réseau ;
- mtu NUMBER : change la MTU (maximal transfert unit) ;
- address LLADDRESS : change l'adresse de l'interface ;
- broadcast LLADDRESS ou brd LLADDRESS : change l'adresse d'émission de couche de lien.

### 1.2 Ip link show

**Ip link show** montre les attributs de l'interface. Abréviations : show, list, lst, sh, ls, l

```
ip link show [ dev NAME ] [ up ]
```

Avec :

- dev NAME : - NAME indique l'interface réseau dont il faut afficher les attributs ;
- up : affiche seulement les interfaces courantes.

### 1.3 Exemples d'utilisation d'ip link set et ip link show

Pour modifier l'adresse de l'interface eth0 :

```
ip link set eth0 address 00 :00 :00 :00 :00 :01
```

Ip link affiche les liens :

```
# ip link show  
  
1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue  
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
```

```
2: eth0: <BROADCAST,MULTICAST,PROMISC,UP> mtu 1500 qdisc pfifo_fast qlen 100
    link/ether 00:06:29:a9:d4:30 brd ff:ff:ff:ff:ff:ff

# ip link show ls eth0

2: eth0: <BROADCAST,MULTICAST,PROMISC,UP> mtu 1500 qdisc pfifo_fast qlen 100
    link/ether 00:06:29:a9:d4:30 brd ff:ff:ff:ff:ff:ff
```

Le nombre écrit au début est le numéro de l'interface, il est suivi par le nom de cette interface. Puis nous avons les autres attributs de l'interface comme le type et la taille de sa file d'attente, ainsi que son adresse.

## 2. LA COMMANDE IP ADDR

### 2.1 Ip address add

**Ip address add** : ajoute une nouvelle adresse. Abréviations : address, addr, a

```
ip addr add [ dev NAME ] [ local ADDRESS ] [ broadcast ADDRESS ]
            [ label NAME ] [ scope SCOPE_VALUE ]
```

Avec :

- **dev NAME** : NAME indique l'interface réseau sur laquelle ajouter l'adresse ;
- **local ADDRESS** : désigne l'adresse de l'interface ;
- **broadcast ADDRESS** : adresse d'émission de l'interface ;
- **label NAME** : chaque adresse est liée à un label ;
- **scope SCOPE\_VALUE** : le secteur où l'adresse est valide Les différentes valeurs sont :
  - global** : l'adresse est globalement valide
  - site** : l'adresse est placée localement, elle est valide à l'intérieur de « site »
  - link** : l'adresse est un lien local, elle est valide seulement sur le dispositif
  - host** : l'adresse est valide seulement à l'intérieur du centre serveur

### 2.2 Ip address del

**Ip address delete** : efface une adresse. Abréviations : delete, del, d.

```
ip addr del [ dev NAME ] [ local ADDRESS ] [ broadcast ADDRESS ]
            [ label NAME ] [ scope SCOPE_VALUE ]
```

Les paramètres sont les mêmes que pour *ip address add*.

### 2.3 Ip address show

**Ip address show**: visualisation des adresses

Abréviations: show, list, lst, sh, ls, l.

```
ip addr show [ dev NAME ] [label PATTERN ] [ {primary| secondary}]
            [ scope SCOPE_VALUE ]
```

Avec :

- **dev NAME** : NAME indique l'interface réseau ;
- **label PATTERN** : affiche seulement les adresses du label indiqué ;
- **primary et secondary** : affiche seulement les adresses primaires ou secondaires ;
- **scope SCOPE\_VALUE** : affiche seulement les adresses du secteur indiqué.

## 2.4 Exemples d'utilisation d'ip addr add et ip addr show

On ajoute l'adresse 192.168.1.2 à l'interface eth0 avec le label eth0 :0 :

```
# ip addr add local 192.168.1.2/24 brd 192.168.1.255 dev eth0 label eth0:0 scope link
```

Puis on visualise l'état des interfaces :

```
# ip addr show

1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 brd 127.255.255.255 scope host lo
2: eth0: <BROADCAST,MULTICAST,PROMISC,UP> mtu 1500 qdisc pfifo_fast qlen 100
    link/ether 00:06:29:a9:d4:30 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.2/24 brd 192.168.1.255 scope link eth0:0
    inet 192.168.0.2/24 brd 192.168.0.255 scope global eth0
```

Notre interface eth0 possède désormais deux adresses MAC, une pour chaque réseau.

## 3. LA COMMANDE IP ROUTE

Abréviations : route, ro, r.

*ip route add*: ajout d'une nouvelle route  
*ip route change*: changement d'une route  
*ip route replace*: changement ou ajout d'une route

Abréviations: add, a ; change, chg ; replace, repl

```
ip route {add|chg|repl} [ to PREFIXE ] [ tos TOS ] [ table TABLE ]
    [ dev NAME ] [ via ADDRESS ] [ src ADDRESS ]
    [ scope SCOPE_VAL ] [protocol RTPROTO]
```

Avec :

- **to PREFIXE** ou **to TYPE PREFIX** : indique la destination ;
- **tos TOS** ou **dsfield TOS** : le type de service (TOS : Type Of Service) ;
- **table TABLE** : la table où ajouter cette route ;
- **dev NAME** : NAME est le nom de l'interface de sortie ;
- **via ADDRESS** : l'adresse du prochain routeur à traverser ;
- **src ADDRESS** : adresse source ;
- **scope SCOPE\_VAL** ;
- **protocol RTPROTO** : le protocole de routage.

### 3.1 Ip route del et ip route add

*ip route delete* : efface une route. Abréviations : delete, del, d.

```
ip route del [ to PREFIXE ] [ tos TOS ] [ table TABLE ] [ dev NAME ]  
[ via ADDRESS ] [ src ADDRESS ] [ scope SCOPE_VAL ] [protocol RTPROTO]
```

Les paramètres sont les mêmes que pour *ip route add*.

### 3.2 Ip route show

*ip route show* : visualisation des routes. Abréviations : show, list, sh, ls, l.

```
ip route sh [ to SELECTOR ] [ tos TOS ] [ table TABLEID ] [ from SELECTOR ]  
[protocol RTPROTO] [ dev NAME ][ scope SCOPE_VAL ]  
[ via PREFIX ][ src PREFIX ]
```

Avec :

- **to SELECTOR** : montre seulement les routes des paquets allant vers l'adresse **SELECTOR** ;
- **tos TOS or dsfield TOS** : montre seulement les routes qui ont le champ TOS correspondant ;
- **table TABLEID** : affiche les routes de cette table ;
- **from SELECTOR** : affiche les routes des paquets provenant de l'adresse source **SELECTOR** ;
- **protocol RTPROTO** : montre seulement les routes des paquets allant vers l'adresse **SELECTOR** ;
- **scope SCOPE\_VAL** : affiche les routes de ce secteur ;
- **type TYPE** : affiche les routes de ce type ;
- **dev NAME** : montre seulement les routes allant vers cette interface ;
- **via PREFIX** : montre seulement les routes allant vers le router **PREFIX** ;
- **src PREFIX** : affiche les routes venant du router **PREFIX**.

### 3.3 Exemples d'utilisation d'ip route show, ip route add et ip route del

Affichage de nos routes :

```
# ip route show  
  
192.168.0.0/24 via 192.168.0.2 dev eth0 scope link  
192.168.0.0/24 dev eth0 scope link  
127.0.0.0/8 dev lo scope link  
default via 192.168.0.1 dev eth0  
  
# ip addr add local 192.168.1.2/24 brd 192.168.1.255 dev eth0 label eth0:0 scope  
link  
  
# ip route show  
  
192.168.1.0/24 dev eth0 proto kernel scope link src 192.168.1.2  
192.168.0.0/24 via 192.168.0.2 dev eth0 scope link  
192.168.0.0/24 dev eth0 scope link  
127.0.0.0/8 dev lo scope link  
default via 192.168.0.1 dev eth0
```

Nous remarquons qu'après avoir exécuté la commande *ip addr add*, la route des paquets dont la destination est le réseau 192.168.1.0 (1<sup>ère</sup> ligne) a été ajoutée. Cette ligne indique que ces paquets passent par l'interface eth0 dont l'adresse est 192.168.1.2, ce qui correspond, en fait, à eth0:0.

**Ajout d'une route : essayons d'ajouter l'adresse de eth0 :0 avec la commande *ip route add* :**

```
# ifconfig eth0:0 192.168.1.2  
# ip route add 192.168.0.2 dev eth0:0  
Cannot find device "eth0:0"
```

**Mais *ip route* ne reconnaît pas l'interface eth0:0.**

**Effacement d'une route :**

```
# ip route del 192.168.1.0/24 dev eth0 proto kernel scope link src 192.168.1.3
```