

# DEMO 1 : CONFIGURER DYNAMIQUEMENT UNE INTERFACE RESEAU ETHERNET

## LA COMMANDE IP

L'objet link et la commande show permettent d'afficher la liste des interfaces réseau connues du système, avec leurs caractéristiques de niveau liaison :

```
# ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enp38s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel
state UP mode DEFAULT group default qlen 1000
    link/ether e4:11:5b:50:13:32 brd ff:ff:ff:ff:ff:ff
3: wlo1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue
state DOWN mode DORMANT group default qlen 1000
    link/ether 56:d0:6d:d0:b9:ab brd ff:ff:ff:ff:ff:ff permaddr
9c:b7:0d:bb:2b:67
```

- Ce système dispose de la pseudo-interface de bouclage lo, ainsi que de deux cartes d'interface réseau : enp38s0 (Ethernet) et wlo1 (Wi-Fi), toutes les deux actives (state UP).

## LA COMMANDE IP ADDRESS

**Afficher les informations d'adresse de toutes les cartes d'interface réseau :**

```
# ip a show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
default
qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp38s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
UP
group default qlen 1000
    link/ether e4:11:5b:50:13:32 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.4/24 brd 192.168.0.255 scope global dynamic noprefixroute
enp38s0
        valid_lft 24479sec preferred_lft 24479sec
    inet6 fe80::5508:86a9:e923:75b3/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: wlo1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP
group
default qlen 1000
    link/ether 9c:b7:0d:bb:2b:67 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.5/24 brd 192.168.0.255 scope global dynamic noprefixroute
wlo1
        valid_lft 24481sec preferred_lft 24481sec
```

```

    inet6 2a01:e35:2439:1510:be93:7dd3:8725:3060/64 scope global dynamic
noprofixroute
    valid_lft 86025sec preferred_lft 86025sec
    inet6 fe80::7f4c:37c9:66fd:4292/64 scope link noprofixroute
    valid_lft forever preferred_lft forever
4: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state
DOWN
group default qlen 1000
    link/ether 52:54:00:a2:5f:41 brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
    valid_lft forever preferred_lft forever
5: virbr0-nic: <BROADCAST,MULTICAST> mtu 1500 qdisc fq_codel master virbr0
state
DOWN group default qlen 1000
    link/ether 52:54:00:a2:5f:41 brd ff:ff:ff:ff:ff:ff

```

- Le système semble avoir de nombreuses cartes d'interface. En réalité, il dispose d'une carte Ethernet enp38s0 (n°2) et d'une interface Wi-Fi wlo1 (n°3). Chacune gère deux adresses utiles, en IPv4 et IPv6. L'interface n°1, lo, est une pseudo-interface de bouclage interne (loopback). Enfin, les deux interfaces n°4 et 5, virbr0 et virbr0-nic ne sont utilisées que par les logiciels de virtualisation (elles sont créées par la bibliothèque libvirt).

#### Pour ne visualiser que l'interface réseau enp38s0 :

```

# ip a show enp38s0
2: enp38s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
UP
group default qlen 1000
    link/ether e4:11:5b:50:13:32 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.108/24 brd 192.168.1.255 scope global dynamic
noprofixroute enp38s0
    valid_lft 40424sec preferred_lft 40424sec
    inet6 2a01:e0a:32c:9940:e611:5bff:fe50:1332/64 scope global dynamic
noprofixroute
    valid_lft 86327sec preferred_lft 86327sec
    inet6 fe80::e611:5bff:fe50:1332/64 scope link noprofixroute
    valid_lft forever preferred_lft forever

```

- La commande affiche les caractéristiques de la carte d'interface réseau, active (state UP), parmi lesquelles :

- Son adresse MAC : e4:11:5b:50:13:32
- Son adresse IPv4 CIDR : 192.168.1.108/24
- Son adresse IPv6 CIDR : fe80::e611:5bff:fe50:1332/64 (adresse de lien local)

#### On peut utiliser le format d'affichage abrégé avec l'option -br :

```

# ip -br a show enp38s0
enp38s0          UP          192.168.1.108/24
2a01:e0a:32c:9940:e611:5bff:fe50:1332/64 fe80::e611:5bff:fe50:1332/64

```

- La carte est active et elle a deux adresses IP, une en version 4 et l'autre en version 6.

#### Définir une nouvelle adresse IPv4 :

Une carte d'interface pouvant gérer plusieurs adresses, IPv4 et/ou IPv6, on veut ajouter l'adresse IPv4 10.1.0.5 à une carte d'interface réseau Ethernet, avec un masque de sous-réseau sur 16 bits (sous-réseau 1 du réseau 10).

**On affiche les interfaces Ethernet disponibles et leurs adresses IPv4 :**

```
# ip -br a show
lo                UNKNOWN      127.0.0.1/8
enp38s0           UP           192.168.1.108/24
virbr0            DOWN        192.168.122.1/24
```

**On ajoute l'adresse souhaitée à la carte Ethernet :**

```
# ip a add 10.1.0.5/16 dev enp38s0
```

**On vérifie :**

```
# ip -br -4 a show dev enp38s0
enp38s0           UP           192.168.1.108/24 10.1.0.5/16
```

□ La carte gère désormais deux adresses IPv4.

**On teste une communication vers une adresse existante sur le sous-réseau 10.1 :**

```
# ping 10.1.0.39
PING 10.1.0.39 (10.1.0.39) 56(84) bytes of data.
64 bytes from 10.1.0.39: icmp_seq=1 ttl=64 time=0.401 ms
64 bytes from 10.1.0.39: icmp_seq=2 ttl=64 time=0.192 ms
64 bytes from 10.1.0.39: icmp_seq=3 ttl=64 time=0.228 ms
64 bytes from 10.1.0.39: icmp_seq=4 ttl=64 time=0.220 ms
^C
--- 10.1.0.39 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 72ms
rtt min/avg/max/mdev = 0.192/0.260/0.401/0.083 ms
```

**Le test effect**

**ué, on décide de supprimer cette adresse :**

```
# ip a del 10.1.0.5/16 dev enp38s0
```

**On vérifie :**

```
# ip -br -4 a show dev enp38s0
enp38s0           UP           192.168.1.108/24
```