

Théorie

Voici quelques sites qui vous aideront à comprendre l'IPv6:

- <http://mirrors.deepspace6.net/Linux+IPv6-HOWTO-fr/>,
- <http://www.e-glop.net/howtos/ipv6-complet.txt>,
- http://www.olympus-zone.net/page_1033_en_Blue.html.

IPv6 chez OVH

Pour nos installations, nous disposons de la /32 suivante: 2001:41D0::/32

Sur le routage, nous utilisons des routeurs Cisco 6k (6509 avec des cartes sup720 qui routent nativement les packets IPv6). Chaque serveur dédié chez OVh dispose donc d'une ipv4 fixe, de une ou plusieurs IP v4 failover et de 18.446.744.073.709.551.616 IPv6 qui sont reliaer a l' IPv4 fixe du serveur.

Un serveur dédié peut ainsi:

- avoir plusieurs certificats SSL
- avoir plusieurs ftp anonymes
- faire de l'hébergement web en IP
- avoir un VPN public
- et autres

Préparation

- le serveur doit être sur un kernel ipv6. Ce kernel est disponible sur le Netboot dans le manager
- il faut récupérer l'ipv6 attribué au serveur dans le manager

Un exemple ipv6

Une IPv4 :

213.186.35.9/24

devient l' IPv6 suivant:

2001:41d0:1:209::/64

Voici des exemples d'IPv6 que l'on peut configurer sur ce serveur dédié :

2001:41d0:1:209::1/64
 2001:41d0:1:209:FF:FF:FF:FF/64
 2001:41d0:1:209:A::1/64
 2001:41d0:1:209::1:B:F/64
 2001:41d0:1:209:1:1:1:1/64

Les écritures suivantes sont équivalentes :

2001:41d0:1:209::1:B:F/64
 2001:41d0:1:209:0:1:B:F/64
 2001:41D0:0001:0209:0000:0001:000B:0000F/64

Mais on ne peut placer qu'une seule fois les ::

Un exemple ifconfig

Après avoir passer le serveur sur un kernel IPv6 après le reboot il apparait déjà une information IPv6 dans le ifconfig:

```
adr inet6: fe80::2e0:4cff:fe99:88d0/64 Scope:Lien
```

Il s'agit ici d'un lien local que le routeur as attribué a ce serveur.
 On peut donc présumer que Ipv6 est opérationnel sur le réseau.

Route: Gateway IPv6

Le routeur (gateway par défaut) pour chaque IPv6 se trouve toujours sur IP:v:6:FF:FF:FF:FF:FF:FF
 Un exemple:

L' IPv6 du serveur: 2001:41D0:1:46e::/64 devient 2001:41D0:1:4 + 5 fois FF.

Gateway ipv6: 2001:41D0:1:4FF:FF:FF:FF:FF

L' IPv6 du serveur: 2001:41d0:1:209::/64 devient donc 2001:41d0:1:2 + 5 fois FF.

Gateway ipv6: 2001:41d0:1:2FF:FF:FF:FF:FF

Installation IPv6

Pour tout les releases linux dans les exemples si dessous il faut placer le serveur sur un kernel ipv6 et il faut récupérer l' IPv6 dans le manager:

Un **uname -a** retourne le bien un kernel IPv6 et la commande **ifconfig** nous montre que grâce au routeur nous avons déjà le lien local en IPv6.

Redhat, CentOS, Fedora

Le serveur dédié que nous allons configurer est en IPv4 : 213.186.35.9/24 ce qui donne IPv6 : 2001:41d0:1:209::/64. Les exemples ci dessous sont identique pour redhat 7.2, fedora et CentOS. Ils peuvent donc aussi être appliquer sur la release plesk 8 ovh fedora 4.

```
ifconfig
```

```
eth0 Lien encap:Ethernet HWaddr 00:E0:4C:99:88:D0
```

```
inet adr:213.186.35.9 Bcast:213.186.35.255 Masque:255.255.255.0
```

```
adr inet6: fe80::2e0:4cff:fe99:88d0/64 Scope:Lien
```

On va ajouter 2001:41D0:1:209::1/64 et 2001:41D0:1:209::5/64 sur eth0:

```
/sbin/ifconfig eth0 inet6 add 2001:41D0:1:209::1/64
```

```
/sbin/ifconfig eth0 inet6 add 2001:41D0:1:209::5/64
```

Afin que l' IPv6 ne disparaît pas au prochain reboot, il faut l'ajouter dans le fichier de configuration de l'interface eth0. On va faire une copie du fichier et puis l'éditer:

```
cp /etc/sysconfig/network-scripts/ifcfg-eth0 /etc/sysconfig/network-scripts/ifcfg-eth.backup
```

```
pico /etc/sysconfig/network-scripts/ifcfg-eth0
```

Voici le contenu du fichier après modification. Les options rouges ont été ajouté:

```
DEVICE=eth0
```

```
BOOTPROTO=static
```

```
IPADDR=213.186.35.9
```

```
NETMASK=255.255.255.0
```

```
ONBOOT=yes
```

```
GATEWAY=213.186.35.254
```

```
IPV6INIT=yes
```

```
IPV6ADDR="2001:41D0:1:209::5/64"
```

```
IPV6ADDR_SECONDARIES="2001:41D0:1:209::5/64"
```

Toutes les IPv6 suivant doivent être sur la ligne IPV6ADDR_SECONDARIES

Il faut encore éditer le fichier /etc/sysconfig/network et ajouter la ligne NETWORKING_IPV6=yes

```
# This line commented out by System Configurator
```

```
NETWORKING=yes
```

```
HOSTNAME=ghost.ovh.net
```

```
GATEWAY=0.0.0.0
```

```
NETWORKING_IPV6=yes
```

Il faut relancer avec /etc/init.d/network restart

Note: Il est possible que redhat affiche une erreur: modprobe: Can't open dependencies file ...Vous pouvez l'ignorer elle influence pas le résultat.

ifconfig ?

```
ifconfig
```

```
eth0 Lien encap:Ethernet HWaddr 00:E0:4C:99:88:D0
```

```
inet adr:213.186.35.9 Bcast:213.186.35.255 Masque:255.255.255.0
```

```
adr inet6: 2001:41d0:1:209::1/64 Scope:Global
```

```
adr inet6: 2001:41d0:1:209::5/64 Scope:Global
```

```
adr inet6: fe80::2e0:4cff:fe99:88d0/64 Scope:Lien
```

Il est possible que vous arriviez à pinguer la gw mais sans pouvoir utiliser votre ipv6, dans ce cas ajoutez une route par défaut :

```
route -A inet6 add default gw votre.gateway.ipv6 dev eth0
```

Par exemple :

```
route -A inet6 add default gw 2001:41d0:1:2FF:FF:FF:FF:FF dev eth0
```

Gentoo

Le serveur dédié que nous allons configurer est en IPv4 : 213.186.40.86/24 ce qui donne IPv6 :

2001:41d0:1:756::1/64 . Vous de votre coté devez récupérer votre IPv6 dans votre manager et booter sur le kernel IPv6. La commande **ifconfig** nous montre que grâce au routeur nous avons déjà le lien local en IPv6.

```
ifconfig
```

```
eth0 Lien encap:Ethernet HWaddr 00:E0:4C:8D:73:4D
```

```
inet adr:213.186.40.86 Bcast:213.186.40.255 Masque:255.255.255.0
```

```
adr inet6: fe80::2e0:4cff:fe8d:734d/64 Scope:Lien
```

Il faut éditer le fichier: /etc/conf.d/net et simplement ajouter les IPv6 dans les () de l'interface eth0. Voici le contenu du fichier après l'ajout des IPv6. Les modifications apporté sont en rouge:

```
config_eth0=( "213.186.40.86 netmask 255.255.255.0"
```

```
"2001:41D0:1:756::1/64"
```

```
" 2001 : 41D0 : 1 : 756 : : 2 / 64 " )
```

```
routes_eth0=( "default gw 213.186.40.254" )
```

On relance l'interface:
/etc/init.d/net.eth0 restart

ifconfig ?

```
ifconfig
eth0 Lien encap:Ethernet HWaddr 00:E0:4C:8D:73:4D
  inet adr:213.186.40.86 Bcast:213.186.40.255 Masque:255.255.255.0
  adr inet6: 2001:41d0:1:756::1/64 Scope:Global
  adr inet6: 2001:41d0:1:756::2/64 Scope:Global
  adr inet6: fe80::2e0:4cff:fe8d:734d/64 Scope:Lien
```

Il est possible que vous arriviez à pinguer la gw mais sans pouvoir utiliser votre ipv6, dans ce cas ajoutez une route par défaut :

```
route -A inet6 add default gw votre.gateway.ipv6 dev eth0
```

Par exemple :

```
route -A inet6 add default gw 2001:41d0:1:2FF:FF:FF:FF:FF dev eth0
```

Debian, Ubuntu

Le serveur dédié que nous allons configurer est en IPv4 : 213.251.132.138/24 ce qui donne IPv6 : 2001:41D0:1:218a::/64. Vous de votre coté devez récupérer votre IPv6 dans votre manager et booter sur le kernel IPv6. La commande **ifconfig** nous montre que grâce au routeur nous avons déjà le lien local en IPv6.

```
eth0 Lien encap:Ethernet HWaddr 00:11:11:6C:E3:93
  inet adr:213.251.132.138 Bcast:213.251.132.255 Masque:255.255.255.0
  adr inet6: fe80::211:11ff:fe6c:e393/64 Scope:Lien
```

Il faut éditer le fichier: /etc/network/interfaces et simplement ajouter les IPv6;

```
iface eth0 inet6 static
  address 2001:41D0:1:218a::1
  netmask 64
```

pour chaque ipv6 en plus il faut ajouter en bas du fichier:

```
post-up /sbin/ifconfig eth0 inet6 add 2001:41D0:1:218a::2/64
pre-down /sbin/ifconfig eth0 inet6 del 2001:41D0:1:218a::2/64
```

puis pour la prochaine:

```
post-up /sbin/ifconfig eth0 inet6 add 2001:41D0:1:218a::4/64
pre-down /sbin/ifconfig eth0 inet6 del 2001:41D0:1:218a::4/64
```

ifconfig ?

ifconfig

```
eth0 Lien encap:Ethernet HWaddr 00:11:11:6C:E3:93
  inet adr:213.251.132.138 Bcast:213.251.132.255 Masque:255.255.255.0
  adr inet6: 2001:41d0:1:218a::1/64 Scope:Global
  adr inet6: 2001:41d0:1:218a::2/64 Scope:Global
  adr inet6: 2001:41d0:1:218a::4/64 Scope:Global
  adr inet6: fe80::211:11ff:fe6c:e393/64 Scope:Lien
```

Il est possible que vous arriviez à pinguer la gw mais sans pouvoir utiliser votre ipv6, dans ce cas ajoutez une route par défaut :

```
route -A inet6 add default gw votre.gateway.ipv6 dev eth0
```

Par exemple :

```
route -A inet6 add default gw 2001:41d0:1:2FF:FF:FF:FF:FF dev eth0
```

Free BSD

Le serveur dédié que nous allons configurer est en IPv4 : 213.186.40.117/24 ce qui donne IPv6 : 2001:41D0:1:775::/64. Nos serveurs Free BSD n'ont pas besoin de booter sur un kernel en netboot, par contre il faut récupérer l'IPv6 dans le manager.

Dans les exemples ici l'interface reseau est rl0. Il est possible que l'interface soit dans votre ifconfig marqué comme vr0. Dans ce cas il suffit tout simplement de remplacer tout les rl0 ci dessous par vr0.

On va commencer par ajouter dans /etc/sysctl.conf les lignes:

```
net.inet6.ip6.accept_rtadv=1
```

et relancer:

```
/etc/rc.d/sysctl restart
```

pour activer l'option router advertisement. Ensuite il suffit d'ajouter la IPv6 dans un premier temps à la main ifconfig rl0 inet6 2001:41D0:1:756::/64

pour obtenir l'info sur le default_router. L'info qu'on cherche est marqué en rouge:

```
rtsol -d rl0
checking if rl0 is ready...
get_llflag() failed, anyway I'll try
send RS on rl0, whose state is 2
received RA from fe80::2d0:3ff:fe75:e000 on rl0, state is 2
stop timer for rl0
there is no timer
```

Toutes les informations pour la configuration de /etc/rc.conf sont la. On va donc ajouter les lignes suivant en bas du fichier:

```

ipv6_enable="YES"
ipv6_network_interface="r10"
ipv6_ifconfig_r10="2001:41D0:1:756::1 prefixlen 64"
ipv6_ifconfig_r10_alias0="2001:41D0:1:756::12 prefixlen 64

```

Chaque IPv6 supplémentaire est ajouter comme pv6_ifconfig_r10_aliasX= .

On relance:

```
/etc/rc.d/network_ipv6 restart
```

ifconfig ?

```
r10: flags=8843 mtu 1500
```

```

inet6 fe80::2e0:4cff:fe89:2883%r10 prefixlen 64 scopeid 0x1
inet 213.186.40.117 netmask 0xfffff00 broadcast 213.186.40.255
inet6 2001:41d0:1:756::1 prefixlen 64 duplicated
inet6 2001:41d0:1:756::12 prefixlen 64
inet6 2001:41d0:1:756::13 prefixlen 64
ether 00:e0:4c:89:28:83

```

Windows

Il faut se connecter via TSE au serveur puis cliquer comme suit sur:

Start > Control Panel > Network Connexion > Local Area Connection > Proprieties

Une fois dans cette fiche faites:

Install > Protocol > Add > Microsoft TCP/IP version 6 > OK

Ensuite:

Start > Command Prompt

pour lancer la commande:

netsh interface ipv6 add address "Local Area Connection" 2001:41d0:1:733::1

Vérification

Test ping6

On ping6 un routeur :

```

ping6 -c 4 2001:41d0:1:2FF:FF:FF:FF:FF
PING 2001:41d0:1:2FF:FF:FF:FF:FF(2001:41d0:1:2ff:ff:ff:ff:ff) from 2001:41d0:1:209::5 : 56 data bytes

```

OVH

64 bytes from 2001:41d0:1:2ff:ff:ff:ff:ff: icmp_seq=0 hops=64 time=127.977 msec
64 bytes from 2001:41d0:1:2ff:ff:ff:ff:ff: icmp_seq=1 hops=64 time=24.242 msec
64 bytes from 2001:41d0:1:2ff:ff:ff:ff:ff: icmp_seq=2 hops=64 time=205.934 msec
64 bytes from 2001:41d0:1:2ff:ff:ff:ff:ff: icmp_seq=3 hops=64 time=129.853 msec

4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/mdev = 24.242/122.001/205.934/64.613 ms

Test traceroute6

Un traceroute6 vers le serveur DNS IPv6 de l'Afnic:

```
traceroute6 2001:660:3006:1::1:1  
traceroute to 2001:660:3006:1
```

```
1:1 (2001:660:3006:1  
1:1) from 2001:41d0:1:209::5, 30 hops max, 16 byte packets  
1 2001:41d0:1:2ff:ff:ff:ff:ff (2001:41d0:1:2ff:ff:ff:ff:ff) 0.726 ms  
0.508 ms *  
2 * * *  
3 renater.sfinx.tm.fr (2001:660:a100:2::1) 0.879 ms 0.787 ms 0.721 ms  
4 afnic.sfinx.tm.fr (2001:660:a100:2::101) 1.13 ms 1.131 ms 1.099 ms  
5 ns3.nic.fr (2001:660:3006:1::1:1) 0.933 ms 0.954 ms 0.922 ms
```

Cela marche donc.

Il y a certains sites qui disposent des outils pour faire un traceroute6 ou un ping6 en ligne.
On peut s'en servir pour vérifier si les IP fraîchement ajoutés fonctionnent.