

IPv6 sur un VPS Arch Linux chez OVH

Stéphane Bortzmeyer

<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 16 septembre 2019. Dernière mise à jour le 30 septembre 2019

<https://www.bortzmeyer.org/ipv6-archlinux-ovh.html>

L'hébergeur OVH a une offre nommée « VPS » (pour "Virtual Private Server" ») qui permet de disposer d'une machine virtuelle connectée à l'Internet, par exemple pour y héberger ses serveurs. Par défaut, on n'a que de l'IPv4 mais IPv6 est possible. Comme je n'ai pas trouvé de documentation qui marche pour le cas où la machine virtuelle tourne sous Arch Linux, je documente ici ma solution, pour que les utilisateurs des moteurs de recherche la trouvent.

Donc, le problème est le suivant. Soit un VPS OVH utilisant le système d'exploitation Arch Linux. Configuré et démarré, il n'a qu'une adresse IPv4 ce qui, en 2019, est inacceptable. OVH permet de l'IPv6, mais ce n'est pas configuré automatiquement par leur système « Cloud Init ». Il faut donc le faire soi-même. OVH documente la façon de faire <<https://docs.ovh.com/fr/vps/configurer-ipv6>> mais Arch Linux n'y est pas mentionné. Du côté de ce système d'exploitation, IPv6 <https://wiki.archlinux.org/index.php/IPv6#Static_address> est documenté (il faut utiliser NetCtl <<https://wiki.archlinux.org/index.php/Netctl>>). A priori, c'est simple, en utilisant les deux documentations, on devrait y arriver. Sauf que cela n'a pas marché, la machine, après redémarrage, n'a toujours pas d'adresse IPv6, et qu'il n'y a pas d'information de débogage disponible. Pas moyen de savoir où je me suis trompé.

Après une heure d'essais infructueux, je suis donc passé à une méthode simple et qui marche immédiatement : écrire un script shell de deux lignes et le faire exécuter au démarrage de la machine. Le script était simplement :

```
ip -6 addr add 2001:41d0:302:2200::180/128 dev eth0
ip -6 route add 2001:41d0:302:2200::1 dev eth0
ip -6 route add default via 2001:41d0:302:2200::1
```

Cette façon de faire est un peu bizarre, avec ce préfixe de longueur 128 (normalement, cela devrait être 64), qui oblige à mettre une route vers... le routeur, mais c'est cohérent avec ce qui est fait par défaut pour IPv4. Une solution plus propre aurait été :

```
ip -6 addr add 2001:41d0:5fa1:b49::180/64 dev eth0
ip -6 route add default via 2001:41d0:5fa1:b49::1
```

Et cela marche mais je ne suis pas sûr que cela permette de joindre les machines du même réseau.

Les adresses IP à utiliser (votre machine, et le routeur par défaut) sont obtenues via l'espace client ("*manager*") de votre compte OVH, rubrique Serveur [Caractère Unicode non montré ¹] VPS [Caractère Unicode non montré] IP.

Une fois ce script écrit, stocké en `/usr/local/sbin/start-ipv6`, et rendu exécutable, il reste à le lancer au démarrage. Pour cela, je me sers de `systemd`. Je crée un fichier `/etc/systemd/system/ipv6.service`. Il contient :

```
[Unit]
Description=Starts IPv6
After=systemd-networkd.service
Before=network.target

[Service]
ExecStart=/bin/true
ExecStartPost=/usr/local/sbin/start-ipv6

[Install]
WantedBy=multi-user.target
```

J'active ce service avec `systemctl enable ipv6` et, au redémarrage de la machine, tout va bien et on a IPv6.

Sinon, Gilles Hamel me dit que ça marche pour lui avec Netplan <<https://wiki.ubuntu.com/Netplan/Design>> (qui est en paquetage sur ArchLinux) et une configuration du genre :

```
% cat /etc/netplan/99-ipv6.yaml
network:
  version: 2
  ethernets:
    eth0:
      addresses:
        - 2001:41d0:5fa1:b49::180/64
      gateway6: 2001:41d0:5fa1:b49::1
```

Mais je n'ai pas testé.

1. Car trop difficile à faire afficher par \LaTeX