

RFC 3315 : Dynamic Host Configuration Protocol for IPv6 (DHCPv6)

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 23 Janvier 2008. Dernière mise à jour le 25 Janvier 2008

Date de publication du RFC : Juillet 2003

<http://www.bortzmeyer.org/3315.html>

DHCP est certainement un des plus grands succès de l'IETF. Le DHCP d'IPv4 est mis en œuvre dans tous les systèmes et présent dans presque tous les réseaux locaux. Mais, dans le monde IPv6, l'arrivée de DHCP est plus récente, et il est concurrencé par un autre système, l'autoconfiguration sans état.

DHCP permet à une machine (qui n'est pas forcément un ordinateur) d'obtenir une adresse IP (ainsi que plusieurs autres informations de configuration) à partir d'un serveur DHCP du réseau local. C'est donc une configuration « avec état », qui nécessite un serveur, par opposition aux systèmes « sans état », comme l'autoconfiguration du RFC 4862¹ qui ne dépendent pas d'un serveur (cette autoconfiguration sans état peut être utilisée à la place de, ou bien en plus de DHCP). Deux utilisations typiques de DHCP sont le SoHo où le routeur ADSL est également serveur DHCP pour les trois PC connectés et le réseau local d'entreprise où deux ou trois machines Unix distribuent adresses IP et informations de configuration à des centaines de machines.

Le DHCP spécifié par notre RFC ne fonctionne que pour IPv6, le RFC 2131 traitant d'IPv4.

DHCP fonctionne par diffusion restreinte. Un **client** DHCP, c'est-à-dire une machine qui veut obtenir une adresses, diffuse (DHCP fonctionne au dessus d'UDP) sa demande à l'adresse "*multicast*" locale au lien FF02::1:2. Le serveur se reconnaît et lui répond. S'il n'y a pas de réponse, c'est, comme dans le DNS, au client de réémettre (section 14).

1. Pour voir le RFC de numéro NNN, <http://www.ietf.org/rfc/rfcNNN.txt>, par exemple <http://www.ietf.org/rfc/rfc4862.txt>

Le serveur choisit sur quels critères il alloue les adresses IP. Il peut les distribuer de manière statique (une même machine a toujours la même adresse IP) ou bien les prendre dans une "pool" d'adresses et chaque client aura donc une adresse « dynamique ». Le fichier de configuration ci-dessous montre un mélange des deux approches.

Il faut bien noter (et notre RFC le fait dans sa section 23) que DHCP n'offre aucune sécurité. Comme il est conçu pour servir des machines non configurées, sur lesquelles on ne souhaite pas intervenir, authentifier la communication est difficile. Un serveur DHCP pirate, ou, tout simplement, un serveur DHCP accidentellement activé, peuvent donc être très gênants. Les approches suggérées dans la section 21 ou dans la section 23 sont en général très complexes (comme l'IPsec du RFC 4301) et ne sont typiquement pas déployées.

Outre l'adresse IP, DHCP peut indiquer des options comme les adresses des serveurs DNS à utiliser (RFC 3646).

Notre version IPv6 de DHCP est assez différente de la version IPv4 (et le RFC est trois fois plus long). Par exemple, l'échange « normal » entre client et serveur prend quatre paquets IP (section 1.3) et non pas deux, l'encodage est très différent, il y a des nouveautés comme l'IA ("*Identity Association*") de la section 10, etc. Il y a aussi des différences visibles à l'utilisateur comme le concept de DUID ("*DHCP Unique Identifier*"), section 9, qui remplace les anciens "*client identifier*" et "*server identifier*" de DHCP v4. Les différences sont telles que le RFC précise que leur intégration n'est pas envisagée.

À l'heure actuelle, il semble qu'il n'existe qu'une seule mise en œuvre de DHCPv6, celle de l'ISC, à partir de la version 4.0. Le fichier de configuration ressemble beaucoup à ce qu'il est en v4 :

```
subnet6 2001:DB8:DEAD:BABE::/64 {
    range6      2001:DB8:DEAD:BABE::100 2001:DB8:DEAD:BABE::FFF;
    # On peut aussi utiliser préfixe/longueur au lieu d'indiquer les
    # adresses de début et de fin de la plage
}
```

On doit lancer le serveur avec l'option -6 (le même démon ne peut pas servir le v4 et le v6 en même temps, les deux protocoles étant trop différents) :

```
# dhcpd -6 -d -f
Internet Systems Consortium DHCP Server 4.0.0
...
Listening on Socket/eth0/2001:db8:dead:babe::/64
Sending on   Socket/eth0/2001:db8:dead:babe::/64

[Puis arrive une requête]

Solicit message from fe80::219:b9ff:fee4:25f9 port 546, transaction ID 0x4BB14F00
Picking pool address 2001:db8:dead:babe::fbb
Sending Advertise to fe80::219:b9ff:fee4:25f9 port 546
Request message from fe80::219:b9ff:fee4:25f9 port 546, transaction ID 0x46B10900
Sending Reply to fe80::219:b9ff:fee4:25f9 port 546
```

(Le concept de "*transaction ID*" est décrit sections 6 et 15.1.) La requête est émise depuis une adresse **lien-local** (ici `fe80::219:b9ff:fee4:25f9`) pas depuis une adresse « tout zéro » comme en IPv4 (section 16 du RFC). Vu avec tcpdump, la requête est :

```
15:07:43.455918 IP6 fe80::219:b9ff:fee4:25f9.546 > ff02::1:2.547: dhcp6 solicit
15:07:43.456098 IP6 fe80::219:b9ff:fee4:2987.547 > fe80::219:b9ff:fee4:25f9.546: dhcp6 advertise
15:07:44.512946 IP6 fe80::219:b9ff:fee4:25f9.546 > ff02::1:2.547: dhcp6 request
15:07:44.513233 IP6 fe80::219:b9ff:fee4:2987.547 > fe80::219:b9ff:fee4:25f9.546: dhcp6 reply
```

On note que l'échange a été celui à quatre messages (Solicit-Advertise-Request-Reply), décrit section 1.3. Le serveur n'a pas répondu directement avec un Reply, parce que le client n'a pas inclus l'option RapidCommit (section 22.14). Cette option n'est pas actuellement gérée par le client DHCP utilisé (l'option `dhcp6.rapid-commit` existe mais la documentation précise qu'elle est ignorée).

Actuellement, l'attribution d'adresses statiques à une machine, en la reconnaissant, par exemple, à son adresse MAC est plus délicate (merci à Shane Kerr pour son aide). Il faut trouver le "*client identifier*" (section 22.2 du RFC, deux méthodes possibles pour le trouver sont expliquées plus loin) et le mettre dans `dhcpd.conf` :

```
host lilith {
    host-identifiant option dhcp6.client-id 0:1:0:1:47:96:21:f7:0:19:b9:e4:25:f9;
    fixed-address6 2001:DB8:DEAD:BABE::2;
}
```

et cette adresse IP fixe est donnée au client.

Pour trouver le "*client identifier*", une méthode spécifique au client DHCP de l'ISC est de regarder dans le fichier des baux du client (typiquement `/var/db/dhclient6.leases`) :

```
...
option dhcp6.client-id 0:1:0:1:47:96:21:f7:0:19:b9:e4:25:f9;
```

Il suffit alors de le copier-coller.

Une autre méthode, plus complexe, mais qui marche avec tous les clients DHCP est de lancer `tcpdump` en mode bavard :

```
# tcpdump -n -vvv ip6 and udp and port 547
12:24:15.084006 IP6 (hlim 64, next-header UDP (17) payload length: 60) fe80::219:b9ff:fee4:25f9.546 > ff02::1:2.
```

Le "*client identifier*", le DUID, se fabrique en concaténant le type de DUID (ici, 1, "*Link-layer Address Plus Time*", section 9.2 du RFC), le type de matériel (1 pour Ethernet), le temps (ici 1201021431, notons que la version actuelle du client DHCP viole le RFC en comptant les secondes à partir de 1970 et pas de 2000) et l'adresse MAC, ce qui redonne le même résultat au prix de quelques calculs avec `bc`.