Dnsmasq - Un serveur DHCP et cache DNS poids-plume

Cf la page officielle de man en français : http://www.linuxcertif.com/man/8/dnsmasq/ reproduite ci-dessous.

Description

dnsmasq est un serveur DHCP et DNS à faible empreinte mémoire. Il offre à la fois les services DNS et DHCP pour un réseau local (LAN).

Dnsmasq accepte les requêtes DNS et y réponds soit en utilisant un petit cache local, soit en effectuant une requête à un serveur DNS récursif externe (par exemple celui de votre fournisseur d'accès internet). Il charge le contenu du fichier /etc/hosts afin que les noms locaux n'apparaissant pas dans les DNS globaux soient tout de même résolus, et assure également la résolution de nom pour les hôtes présents dans le service DHCP.

Le serveur DHCP Dnsmasq DHCP supporte les définitions d'adresses statiques et les réseaux multiples. Il envoie par défaut un jeu raisonnable de paramètres DHCP, et peut être configuré pour envoyer n'importe quel option DHCP. Il inclut un serveur TFTP sécurisé en lecture seule permettant le démarrage via le réseau/PXE de clients DHCP et supporte également le protocole BOOTP.

Dnsmasq supporte IPv6 pour le DNS mais pas pour le DHCP.

Options

Il est possible d'utiliser des options sans leur donner de paramètre. Dans ce cas, la fonction correspondante sera désactivée. Par exemple –pid-file= (sans paramètre après le =) désactive l'écriture du fichier PID. Sur BSD, à moins que le logiciel ne soit compilé avec la bibliothèque GNU getopt, la forme longue des options ne fonctionne pas en ligne de commande; Elle est toujours supportée dans le fichier de configuration.

no-hosts
: Ne pas charger les noms du fichier **/etc/hosts**

addn-hosts=<fichier>
: Fichiers d'hôtes additionnels

- Lire le fichier <fichier> en plus de **/etc/hosts**
- Cette option peut être répétée pour ajouter d'autres fichiers.

- Si c'est un nom de répertoire qui est donné, lit les fichiers contenus dans ce répertoire.

expand-hosts
: Ajoute le nom de domaine aux noms simples (ne contenant pas de point dans le nom)
  - contenus dans le fichier **/etc/hosts**
  - et pour le service DHCP

- cela ne s'applique pas au nom de domaine dans les CNAME, les enregistrements PTR, TXT, etc...

local-ttl=<durée>
: time-to-live (en secondes) à retourner.

- Lorsque Dnsmasq répond avec une information provenant du fichier **/etc/hosts** ou avec un bail DHCP, il donne un temps de vie (time-to-live) nul pour que la requête ne soit pas mise en cache. C'est généralement le mieux.

neg-ttl=<durée>
: durée de vie par défaut (en secondes) que dnsmasq utilise pour mettre les réponses négatives dans son cache, même en l'absence d'enregistrement SOA.

- Les réponses négatives des serveurs amont contiennent normalement une information de durée de vie (time-to-live) dans les enregistrements SOA, information dont dnsmasq se sert pour mettre la réponse en cache.
- Si la réponse du serveur amont omet cette information, dnsmasq ne met pas la réponse en cache.

max-ttl=<durée>
: valeur maximum de TTL fournie aux clients.

- Cette valeur maximum de TTL sera fournie aux clients en remplacement de la vraie valeur de TTL si cette dernière est supérieure.
- La valeur réelle de TTL est cependant conservée en cache pour éviter de saturer les serveurs DNS en amont.

log-queries
: Enregistrer les résultats des requêtes DNS traitées par Dnsmasq dans un fichier de traces (“logs”). Active la génération d'un état complet du cache lors de la réception d'un signal SIGUSR1.

log-facility=<facility>
: Définit la “facility” dans laquelle Dnsmasq enverra ses entrées syslog, par défaut DAEMON ou LOCAL0 si le mode debug est activé. Si la “facility” contient au moins un caractère “/”, alors Dnsmasq considère qu'il s'agit d'un fichier et enverra les logs dans le fichier correspondant à la place du syslog. Si la “facility” est '-', alors dnsmasq envoie les logs sur la sortie d'erreur standard stderr. (Les erreurs lors de la lecture de la configuration vont toujours vers le syslog, mais tous les messages postérieurs à un démarrage réussi seront exclusivement envoyés vers le fichier de logs). Lorsque Dnsmasq est configuré pour envoyer ses traces vers un fichier, la réception d'un signal

SIGUSR2 entraine la fermeture et réouverture du fichier. Cela permet la rotation de fichiers de traces sans nécessiter l'arrêt de Dnsmasq.

log-async[=<lignes>]
: Permet l'envoi de traces de manière asynchrone, et de manière optionnelle, le nombre de lignes devant être mises dans la file d'attente par Dnsmasq lorsque l'écriture vers le syslog est lente. Dnsmasq peut envoyer ses logs de manière asynchrone : cela lui permet de continuer à fonctionner sans être bloqué par le syslog, et permet à syslog d'utiliser Dnsmasq pour les résolutions DNS sans risque d'interblocage. Si la file d'attente devient pleine, Dnsmasq loggera le dépassement de file et le nombre de messages perdus. La longueur par défaut de la file d'attente est de 5 et une valeur saine sera comprise entre 5 et 25, avec une limite maximum imposée de 100.

pid-file=<chemin>
: Spécifie un fichier dans lequel stocker le numéro de processus (pid). La valeur par défaut est /var/run/dnsmasq.pid.

user=<nom d'utilisateur>
: Spécifie l'identité (nom d'utilisateur) prise par Dnsmasq après le démarrage. Dnsmasq doit normalement être démarré en temps que root (“super-utilisateur”), mais abandonne ses privilèges après le démarrage en changeant d'identité. Normalement cet utilisateur est l'utilisateur nobody (“personne”), mais il est possible d'en définir un autre par le biais de ce paramètre.

group=<nom de groupe>
: Spécifie le groupe sous lequel Dnsmasq s'exécute. Par défaut, il s'agit du groupe “dip”, afin de faciliter l'accès au fichier /etc/ppp/resolv.conf qui n'est en général pas en lecture par tout le monde.

version
: Imprime le numéro de version.

port=<port>
: Ecoute sur le port numéro <port> au lieu du port DNS standard (53). Paramétrer cette valeur à zéro désactive complètement la fonction DNS pour ne laisser actif que le DHCP ou le TFTP.

edns-packet-max=<taille>
: Spécifie la taille maximum de paquet UDP EDNS.0 supporté par le relai DNS. Le défaut est de 4096, qui est la valeur recommandée dans la RFC5625.

query-port=<numéro de port>
: Envoie et écoute les requêtes DNS sortantes depuis le port UDP spécifié par <numéro de port>, et non sur un port aléatoire. NOTE : Cette option rends dnsmasq moins sûr contre les attaques par usurpation DNS (“DNS spoofing”), mais cela peut permettre d'utiliser moins de ressources et d'être plus rapide. Donner une valeur de zéro à cette option restaure le comportement par défaut présent dans les versions de dnsmasq inférieures à 2.43 qui consiste à n'allouer qu'un seul port alloué par le système d'exploitation.

min-port=<port>
: Ne pas utiliser de port dont le numéro est inférieur à la valeur donnée en paramètre pour les requêtes DNS sortantes. Dnsmasq choisis un port source aléatoire pour les requêtes sortantes : lorsque cette option est fournie, les ports utilisés seront toujours au dessus de la valeur spécifiée. Utile pour des systèmes derrière des dispositifs garde-barrières (“firewalls”).

interface=<nom d'interface>
: N'écouter que sur l'interface réseau spécifiée. Dnsmasq aujoute automatiquement l'interface locale (“loopback”) à la liste des interfaces lorsque l'option –interface est utilisée. Si aucune option –interface ou –listen-address n'est donnée, Dnsmasq écoutera sur toutes les interfaces disponibles sauf celle(s) spécifiée(s) par l'option –except-

interface. Les alias d'interfaces IP (e-g “eth1:0”) ne peuvent être utilisés ni avec –interface ni –except-interface. Utiliser l'option –listen-address à la place.

except-interface=<interface name>

Ne pas écouter sur l'interface spécifiée. Notez que l'ordre dans lesquelles les options –listen-address , –interface et –except-interface sont fournies n'importe pas, et que l'option –except-interface l'emporte toujours sur les autres.

no-dhcp-interface=<nom d'interface>

Ne pas fournir de service DHCP sur l'interface spécifiée, mais fournir tout de même le service DNS.

listen-address=<adresse IP>

Ecouter sur la ou les adresse(s) IP spécifiée(s). Les options –interface et –listen-address peuvent-être spécifiées simultanément, auquel cas un jeu d'interfaces et d'adresses seront utilisées. Notez que si aucune option –interface n'est donnée alors qu'une option –listen-address l'est, Dnsmasq n'écoutera pas automatiquement sur l'interface locale (“loopback”). Pour activer l'écoute sur l'interface locale, il est alors nécessaire de fournir explicitement son adresse IP, 127.0.0.1 via l'option –listen-address.

bind-interfaces

Sur les systèmes qui le supporte, Dnsmasq s'associe avec l'interface joker (“wildcard”), même lorsqu'il ne doit écouter que sur certaines interfaces. Par la suite, il rejette les requêtes auxquelles il ne doit pas répondre. Cette situation présente l'avantage de fonctionner même lorsque les interfaces vont et viennent ou changent d'adresses. L'option –bind-interfaces force Dnsmasq à ne réellement s'associer qu'avec les interfaces sur lesquelles il doit écouter. L'un des seuls cas où cette option est utile est celui où un autre serveur de nom (ou une autre instance de Dnsmasq) tourne sur la même machine. Utiliser cette option permet également d'avoir plusieurs instances de Dnsmasq fournissant un service DHCP sur la même machine.

localise-queries

Retourne des réponses aux requêtes DNS dépendantes de l'interface sur laquelle la requête a été reçue, à partir du fichier /etc/hosts. Si un nom dans /etc/hosts a plus d'une adresse associée avec lui, et qu'une des adresses au moins est dans le même sous-réseau que l'interface sur laquelle la requête a été reçue, alors ne retourne que la(les) adresse(s) du sous-réseau considéré. Cela permet d'avoir dans /etc/hosts un serveur avec de multiples adresses, une pour chacune de ses interfaces, et de fournir aux hôtes l'adresse correcte (basée sur le réseau auquel ils sont attachés). Cette possibilité est actuellement limitée à IPv4.

bogus-priv

Fausse résolution inverse pour les réseaux privés. Toutes les requêtes DNS inverses pour des adresses IP privées (ie 192.168.x.x, etc…) qui ne sont pas trouvées dans /etc/hosts ou dans le fichier de baux DHCP se voient retournées une réponse “pas de tel domaine” (“no such domain”) au lieu d'être transmises aux serveurs de nom amont (“upstream server”).

alias=[<ancienne IP>][<IP de début>-<IP de fin>],<nouvelle IP>[,<masque>]

Modifie les adresses IPv4 retournées par les serveurs de nom amont; <ancienne IP> est remplacée par <nouvelle IP>. Si le <masque> optionnel est fourni, alors toute adresse correspondant à l'adresse <ancienne IP>/<masque> sera réécrite. Ainsi par exemple –alias=1.2.3.0,6.7.8.0,255.255.255.0 modifiera 1.2.3.56 en 6.7.8.56 et 1.2.3.67 en 6.7.8.67. Cette fonctionnalité correspond à ce que les routeurs Cisco PIX appellent “bidouillage DNS” (“DNS doctoring”). Si l'ancienne IP est donnée sous la forme d'une gamme d'adresses, alors seules les adresses dans cette gamme seront réecrites, et non le sous-réseau dans son ensemble. Ainsi,

–alias=192.168.0.10-192.168.0.40,10.0.0.0,255.255.255.0 fait correspondre 192.168.0.10→192.168.0.40 à 10.0.0.10→10.0.0.40

bogus-nxdomain=<adresse IP>

Transforme les réponses contenant l'adresse IP fournie en réponses “pas de tel domaine” (“no such domain”). Ceci a pour but de neutraliser la modification sournoise mise en place par Verisign en septembre 2003, lorsqu'ils ont commencé à retourner l'adresse d'un serveur web publicitaire en réponse aux requêtes pour les noms de domaines non enregistrés, au lieu de la réponse correcte “NXDOMAIN”. Cette option demande à Dnsmasq de retourner la réponse correcte lorsqu'il constate ce comportement. L'adresse retournée par Verisign en septembre 2003 est 64.94.110.11.

filterwin2k

Les dernières versions de windows font des requêtes DNS périodiques auxquelles non seulement les serveurs DNS publics ne peuvent donner de réponse, mais qui, de surcroît, peuvent poser des problèmes en déclenchant des connexions intempestives pour des liens réseaux avec des connexions “à la demande”. Fournir cette option active le filtrage des requêtes de ce type. Les requêtes bloquées sont les requêtes pour les entrées de type SOA ou SRV, ainsi que les requêtes de type ANY avec des noms possédant des caractères sous-lignés (requêtes pour des serveurs LDAP).

resolv-file=<fichier>

Lis les adresses des serveurs de nom amont dans le fichier de nom <fichier>, au lieu du fichier /etc/resolv.conf. Pour le format de ce fichier, voir dans le manuel pour resolv.conf(5) les entrées correspondant aux serveurs de noms (nameserver). Dnsmasq peut lire plusieurs fichiers de type resolv.conf, le premier fichier spécifié remplace le fichier par défaut, le contenu des suivants est rajouté dans la liste des fichiers à consulter. Seul le fichier ayant la dernière date de modification sera chargé en mémoire.

no-resolv

Ne pas lire le contenu du fichier /etc/resolv.conf. N'obtenir l'adresse des serveurs de nom amont que depuis la ligne de commande ou le fichier de configuration de Dnsmasq.

enable-dbus

Autoriser la mise à jour de la configuration de Dnsmasq par le biais d'appel de méthodes DBus. Il est possible par ce biais de mettre à jour l'adresse de serveurs DNS amont (et les domaines correspondants) et de vider le cache. Cette option nécessite que Dnsmasq soit compilé avec le support DBus.

strict-order

Par défaut, Dnsmasq envoie les requêtes à n'importe lequel des serveurs amonts dont il a connaissance tout en essayant de favoriser les serveurs qu'il sait fonctionner. Cette option force Dnsmasq à essayer d'interroger, pour chaque requête, les serveurs DNS dans leur ordre d'apparition dans le fichier /etc/resolv.conf.

all-servers

Par défaut, lorsque dnsmasq a plus d'un serveur amont disponible, il n'envoie les requêtes qu'à un seul serveur. Spécifier cette option force dnsmasq à effectuer ses requêtes à tous les serveurs disponibles. Le résultat renvoyé au client sera celui fournit par le premier serveur ayant répondu.

stop-dns-rebind

Rejete (et enregistre dans le journal d'activité) les adresses dans la gamme d'adresses IP privée (au sens RFC1918) qui pourraient être renvoyées par les serveurs amonts suite à une résolution de nom. Cela bloque les attaques cherchant à détourner de leur usage les logiciels de navigation web ('browser') en s'en servant pour découvrir les machines situées sur le réseau local.

rebind-localhost-ok

Exclue 127.0.0/8 des vérifications de réassociation DNS. Cette gamme d'adresses est

retournée par les serveurs Realtime Blackhole (RBL, utilisés dans la lutte contre le spam), la bloquer peut entraîner des dysfonctionnements de ces services.

rebind-domain-ok=[<domaine>]|[[/<domaine>/[<domaine>/]

Ne pas détecter ni bloquer les actions de type dns-rebind pour ces domaines. Cette option peut prendre comme valeur soit un nom de domaine soit plusieurs noms de domains entourés par des '/', selon une syntaxe similaire à l'option –server, c-à-d : –rebind-domain-ok=/domaine1/domaine2/domaine3/

no-poll

Ne pas vérifier régulièrement si le fichier /etc/resolv.conf a été modifié.

clear-on-reload

Lorsque le fichier /etc/resolv.conf est relu, vider le cache DNS. Cela est utile si les nouveaux serveurs sont susceptibles d'avoir des données différentes de celles stockées dans le cache.

domain-needed

Indique à Dnsmasq de ne jamais transmettre en amont de requêtes pour des noms simples, ne comprenant donc ni points ni nom de domaine. Si un nom n'est pas dans /etc/hosts ou dans la liste des baux DHCP, alors une réponse de type “non trouvé” est renvoyée.

local, server=[/[<domaine>]/[domaine/]][<Adresse IP>[#<port>][@<Adresse IP source>|<interface>[#<port>]]]

Spécifie directement l'adresse IP d'un serveur de nom amont. Cette option ne supprime pas la lecture du fichier /etc/resolv.conf, utiliser pour cela l'option -R. Si un ou plusieurs nom(s) de domaine(s) optionnel(s) sont fournis, ce serveur sera uniquement utilisé uniquement pour ce(s) domaine(s), et toute requête concernant ce(s) domaine(s) sera adressée uniquement à ce serveur. Cette option est destinée aux serveurs de nom privés : si vous avez un serveur de nom sur votre réseau ayant pour adresse IP 192.168.1.1 et effectuant la résolution des noms de la forme xxx.internal.thekelleys.org.uk, alors -S /internal.thekelleys.org.uk/192.168.1.1 enverra toutes les requêtes pour les machines internes vers ce serveur de nom, alors que toutes les autres requêtes seront adressées aux serveurs indiqués dans le fichier /etc/resolv.conf. Une spécification de nom de domaine vide, // possède le sens particulier de “pour les noms non qualifiés uniquement”, c'est-à-dire les noms ne possédant pas de points. Un port non standard peut être rajouté à la suite des adresses IP en utilisant le caractère #. Plus d'une option -S est autorisée, en répétant les domaines et adresses IP comme requis.

Le domaine le plus spécifique l'emporte sur le domaine le moins spécifique, ainsi : –server=/google.com/1.2.3.4 –server=/www.google.com/2.3.4.5 enverra les requêtes pour *.google.com à 1.2.3.4, à l'exception des requêtes *www.google.com, qui seront envoyées à 2.3.4.5.

L'adresse spéciale '#' signifie “utiliser les serveurs standards”, ainsi –server=/google.com/1.2.3.4 –server=/www.google.com/# enverra les requêtes pour *.google.com à 1.2.3.4, à l'exception des requêtes pour *www.google.com qui seront envoyées comme d'habitude (c-à-d aux serveurs définis par défaut).

Il est également permis de donner une option -S avec un nom de domaine mais sans adresse IP; Cela informe Dnsmasq que le domaine est local et qu'il doit répondre aux requêtes le concernant depuis les entrées contenues dans le fichier /etc/hosts ou les baux DHCP, et ne doit en aucun cas transmettre les requêtes aux serveurs amonts. local est synonyme de server (“serveur”) afin de rendre plus claire l'utilisation de cette option pour cet usage particulier.

La chaîne de caractères optionnelle suivant le caractère @ permet de définir la source que Dnsmasq doit utiliser pour les réponses à ce serveur de nom. Il doit s'agir d'une des

adresses IP appartenant à la machine sur laquelle tourne Dnsmasq ou sinon la ligne sera ignorée et une erreur sera consignée dans le journal des événements, ou alors d'un nom d'interface. Si un nom d'interface est donné, alors les requêtes vers le serveur de nom seront envoyées depuis cette interface; si une adresse ip est donnée, alors l'adresse source de la requête sera l'adresse en question. L'option query-port est ignorée pour tous les serveurs ayant une adresse source spécifiée, mais il est possible de la donner directement dans la spécification de l'adresse source. Forcer les requêtes à être émises depuis une interface spécifique n'est pas possible sur toutes les plateformes supportées par dnsmasq.

address=/<domaine>/[domaine/]<adresse IP>

Spécifie une adresse IP à retourner pour toute requête pour les domaines fournis en option. Les requêtes pour ce(s) domaine(s) ne sont jamais transmises aux serveurs amonts et reçoivent comme réponse l'adresse IP spécifiée qui peut être une adresse IPv4 ou IPv6. Pour donner à la fois une adresse IPv4 et une adresse IPv6 pour un domaine, utiliser plusieurs options -A. Il faut noter que le contenu du fichier /etc/hosts et de celui des baux DHCP supplante ceci pour des noms individuels. Une utilisation courante de cette option est de rediriger la totalité du domaine doubleclick.net vers un serveur web local afin d'éviter les bannières publicitaires. La spécification de domaine fonctionne de la même façon que –server, avec la caractéristique supplémentaire que /#/ coïncide avec tout domaine. Ainsi,
address=/#/1.2.3.4 retournera 1.2.3.4 pour toute requête n'ayant de réponse ni dans /etc/hosts, ni dans les baux DHCP, et n'étant pas transmise à un serveur spécifique par le biais d'une directive –server.

mx-host=<nom de l'hôte>[[,<nom du MX>],<préference>]

Spécifie un enregistrement de type MX pour <nom de l'hôte> retournant le nom donné dans <nom du MX> (s'il est présent), ou sinon le nom spécifié dans l'option –mx-target si elle est présente. Sinon retourne le nom de la machine sur laquelle Dnsmasq tourne. La valeur par défaut (spécifiée dans l'option –mx-target) est utile dans un réseau local pour rediriger les courriers électroniques vers un serveur central. La valeur de préférence est optionnelle et vaut par défaut 1 si elle n'est pas spécifiée. Plus d'une entrée MX peut être fournie pour un hôte donné.

mx-target=<nom d'hôte>

Spécifie la réponse par défaut fournie par Dnsmasq pour les requêtes sur des enregistrements de type MX. Voir –mx-host. Si –mx-target est donné mais pas de –mx-host, alors Dnsmasq retourne comme réponse un enregistrement MX contenant le nom d'hôte spécifié dans l'option –mx-target pour toute requête concernant le MX de la machine sur laquelle tourne Dnsmasq.

selfmx

Définit, pour toutes les machines locales, un MX correspondant à l'hôte considéré. Les machines locales sont celles définies dans le fichier /etc/hosts ou dans un bail DHCP.

localmx

Définit, pour toutes les machines locales, un enregistrement MX pointant sur l'hôte spécifié par mx-target (ou la machine sur laquelle Dnsmasq tourne). Les machines locales sont celles définies dans le fichier /etc/hosts ou dans un bail DHCP.

srv-host=<_service>.<_protocole>.[<domaine>],[<cible>[,<port>[,<priorité>[,<poids>]]]]

Spécifie un enregistrement DNS de type SRV. Voir la RFC2782 pour plus de détails. Si le champs <domaine> n'est pas fourni, prends par défaut la valeur fournie dans l'option –domain. La valeur par défaut pour le domaine est vide et le port par défaut est 1, alors que les poids et priorités par défaut sont 0. Attention lorsque vous transposez des valeurs issues d'une configuration BIND : les ports, poids et priorités sont dans un ordre différents. Pour un service/domaine donné, plus d'un enregistrement SRV est autorisé et

tous les enregistrements qui coïncident sont retournés dans la réponse.

txt-record=<nom>[[,<texte>],<texte>]

Définit un enregistrement DNS de type TXT. La valeur de l'enregistrement TXT est un ensemble de chaînes de caractères, donc un nombre variable de chaînes de caractères peuvent être spécifiées, séparées par des virgules.

ptr-record=<nom>[,<cible>]

Définit un enregistrement DNS de type PTR.

naptr-record=<nom>,<ordre>,<préférence>,<drapeaux>,<service>,<expr. régulière>[,<remplacement>]

Retourne un enregistrement de type NAPTR, tel que spécifié dans le RFC3403.

cname=<cname>,<cible>

Retourne un enregistrement de type CNAME qui indique que <cname> est en réalité <cible>. Il existe des contraintes significatives sur la valeur de cible; il doit s'agir d'un nom DNS qui est connu de dnsmasq via /etc/hosts (ou un fichier hôtes additionnel) ou via DHCP. Si une cible ne satisfait pas ces critères, le CNAME est ignoré. Le CNAME doit être unique, mais il est autorisé d'avoir plus d'un CNAME pointant vers la même cible.

interface-name=<nom>,<interface>

Définit un entregistrement DNS associant le nom avec l'adresse primaire sur l'interface donnée en argument. Cette option spécifie un enregistrement de type A pour le nom donné en argument de la même façon que s'il était défini par une ligne de /etc/hosts, sauf que l'adresse n'est pas constante mais dépendante de l'interface définie. Si l'interface est inactive, non existante ou non configurée, une réponse vide est fournie. Un enregistrement inverse (PTR) est également créé par cette option, associant l'adresse de l'interface avec le nom. Plus d'un nom peut être associé à une interface donnée en répétant cette option plusieurs fois; dans ce cas, l'enregistrement inverse pointe vers le nom fourni dans la première instance de cette option.

cache-size=<taille>

Définit la taille du cache de Dnsmasq. La valeur par défaut est de 150 noms. Définir une valeur de zéro désactive le cache.

no-negcache

Désactive le “cache négatif”. Le “cache négatif” permet à Dnsmasq de se souvenir des réponses de type “no such domain” fournies par les serveurs DNS en amont et de fournir les réponses sans avoir à re-transmettre les requêtes aux serveurs amont.

dns-forward-max=<nombre de requêtes>

Définit le nombre maximum de requêtes DNS simultanées. La valeur par défaut est 150, ce qui devrait être suffisant dans la majorité des configurations. La seule situation identifiée dans laquelle cette valeur nécessite d'être augmentée est lorsqu'un serveur web a la résolution de nom activée pour l'enregistrement de son journal des requêtes, ce qui peut générer un nombre important de requêtes simultanées.

dhcp-range=[interface:<interface>,][tag:<label>[,tag:<label>],][set:<label],]<adresse de début>,<adresse de fin>[,<masque de réseau>[,<broadcast>]][,<durée de bail>]

Active le serveur DHCP. Les adresses seront données dans la plage comprise entre <adresse de début> et <adresse de fin> et à partir des adresses définies statiquement dans l'option dhcp-host. Si une durée de bail est donnée, alors les baux seront donnés pour cette durée. La durée de bail est donnée en secondes, en minutes (exemple : 45m), en heures (exemple : 1h) ou être la chaine de caractère “infinite” pour une durée indéterminée. Si aucune valeur n'est donnée, une durée de bail par défaut de une heure est appliquée. La valeur minimum pour un bail DHCP est de 2 minutes. Cette option peut être répétée, avec différentes adresses, pour activer le service DHCP sur plus d'un réseau. Pour des réseaux directement connectés (c'est-à-dire des réseaux dans lesquels

la machine sur laquelle tourne Dnsmasq possède une interface), le masque de réseau est optionnel. Il est par contre requis pour les réseaux pour lesquels le service DHCP se fait via un relais DHCP (“relay agent”). L'adresse de broadcast est toujours optionnelle.
Il est toujours possible d'avoir plus d'une plage DHCP pour un même sous-réseau.
L'identifiant de label optionnel set:<label> fournie une étiquette alphanumérique qui identifie ce réseau, afin de permettre la fourniture d'options DHCP spécifiques à chaque réseau. Lorsque préfixé par 'tag:', la signification change, et au lieu de définir un label, il définit le label pour laquelle la règle s'applique. Un seul label peut- être défini mais plusieurs labels peuvent coïncider.
L'adresse de fin peut être remplacée par le mot-clef static (“statique”) qui indique à Dnsmasq d'activer le service DHCP pour le réseau spécifié, mais de ne pas activer l'allocation dynamique d'adresses IP : Seuls les hôtes possédant des adresses IP statiques fournies via dhcp-host ou présentes dans le fichier /etc/ethers seront alors servis par le DHCP.
L'adresse de fin peut-être remplacée par le mot-clef proxy , auquel cas Dnsmasq fournira un service de DHCP proxy pour le sous-réseau spécifié. (voir pxe-prompt et pxe-service pour plus de détails).
La section interface:<nom d'interface> n'est normalement pas utilisée. Se référer aux indications de la section NOTES pour plus de détail à ce sujet.

dhcp-host=[<adresse matérielle>][,id:<identifiant client>|*][,set:<label>][,<adresse IP>][,<nom d'hôte>][,<durée de bail>][,ignore]

Spécifie les paramètres DHCP relatifs à un hôte. Cela permet à une machine possédant une adresse matérielle spécifique de se voir toujours allouée les mêmes nom d'hôte, adresse IP et durée de bail. Un nom d'hôte spécifié comme ceci remplace le nom fourni par le client DHCP de la machine hôte. Il est également possible d'omettre l'adresse matérielle et d'inclure le nom d'hôte, auquel cas l'adresse IP et la durée de bail s'appliqueront à toute machine se réclamant de ce nom. Par exemple –dhcp-host=00:20:e0:3b:13:af,wap,infinite spécifie à Dnsmasq de fournir à la machine d'adresse matérielle 00:20:e0:3b:13:af le nom, et un bail de durée indéterminée. dhcp-host=lap,192.168.0.199 spécifie à Dnsmasq d'allouer toujours à la machine portant le nom lap l'adresse IP 192.168.0.199.

Les adresses allouées de la sorte ne sont pas contraintes à une plage d'adresse spécifiée par une option –dhcp-range, mais elles se trouver dans le même sous-réseau qu'une plage dhcp-range valide. Pour les sous-réseaux qui n'ont pas besoin d'adresses dynamiquement allouées, utiliser le mot-clef “static” dans la déclaration de plage d'adresses dhcp-range.
Il est possible d'utiliser des identifiants clients plutôt que des adresses matérielles pour identifier les hôtes, en préfixant par ceux-ci par 'id:'. Ainsi, –dhcp-host=id:01:02:03:04,..... réfère à l'hôte d'identifiant 01:02:03:04. Il est également possible de spécifier l'identifiant client sous la forme d'une chaîne de caractères, comme ceci : –dhcp-host=id:identifiantclientsousformedechaine,.....
L'option spéciale id:* signifie : “ignorer tout identifiant client et n'utiliser que l'adresse matérielle”. Cela est utile lorsqu'un client présente un identifiant client mais pas les autres.
Si un nom apparaît dans /etc/hosts, l'adresse associée peut être allouée à un bail DHCP mais seulement si une option –dhcp-host spécifiant le nom existe par ailleurs. Seul un nom d'hôte peut-être donné dans une option dhcp-host , mais les alias sont possibles au travers de l'utilisation des CNAMEs. (Voir –cname). Le mot clef “ignore” (“ignorer”) indique à Dnsmasq de ne jamais fournir de bail DHCP à une machine. La machine peut être spécifiée par son adresse matérielle, son identifiant client ou son nom d'hôte. Par exemple –dhcp-host=00:20:e0:3b:13:af,ignore Cela est utile lorsqu'un autre serveur

DHCP sur le réseau doit être utilisé par certaines machines.
Le paramètre set:<identifiant réseau> permet de définir un identifiant de réseau lorsque l'option dhcp-host est utilisée. Cela peut servir à sélectionner des options DHCP juste pour cet hôte. Plus d'un label peut être fourni dans une directive dhcp-host (et dans cette seule directive). Lorsqu'une machine coïncide avec une directive dhcp-host (ou une impliquée par /etc/ethers), alors le label réservé “known” (“connu”) est associé. Cela permet à Dnsmasq d'être configuré pour ignorer les requêtes issus de machines inconnue
par le biais de –dhcp-ignore=tag:!known.
Les adresses ethernet (mais pas les identifiants clients) peuvent être définies avec des octets joker, ainsi par exemple –dhcp-host=00:20:e0:3b:13:*,ignore demande à Dnsmasq d'ignorer une gamme d'adresses matérielles. Il est à noter que “*” doit-être précédé d'un caractère d'échappement ou mis entre guillemets lorsque spécifié en option de ligne de commande, mais pas dans le fichier de configuration.
Les adresses matérielles coïncident en principe avec n'importe quel type de réseau (ARP), mais il est possible de les limiter à un seul type ARP en les précédant du type ARP (en Hexadécimal) et de “-”.
Ainsi –dhcp-host=06-00:20:e0:3b:13:af,1.2.3.4 coïncidera uniquement avec des adresses matérielles Token-Ring, puisque le type ARP pour une adresse Token-Ring est 6.
Un cas spécial correspond à l'inclusion d'une ou plusieurs adresses matérielles, c-à-d : –dhcp-host=11:22:33:44:55:66,12:34:56:78:90:12,192.168.0.2. Cela permet à une adresse IP d'être associé à plusieurs adresses matérielles, et donne à dnsmasq la permission d'abandonner un bail DHCP attribué à l'une de ces adresses lorsqu'une autre adresse dans la liste demande un bail. Ceci est une opération dangereuse qui ne fonctionnera de manière fiable que si une adresse matérielle est active à un moment donné et dnsmasq n'a aucun moyen de s'assurer de cela. Cela est utile, par exemple, pour allouer une adresse IP stable à un laptop qui aurait à la fois une connexion filaire et sans-fil.

dhcp-hostsfile=<fichier>

Lis les informations d'hôtes DHCP dans le fichier spécifié. Le fichier contient des informations à raison d'un hôte par ligne. Le format d'une ligne est la même que le texte fourni à la droite sur caractère “=” dans l'option –dhcp-host. L'avantage de stocker les informations sur les hôtes DHCP dans ce fichier est que celles-ci peuvent être modifiées sans recharger Dnsmasq; le fichier sera relu lorsque Dnsmasq reçoit un signal SIGHUP.

dhcp-optsfile=<fichier>

Lis les informations relatives aux options DHCP dans le fichier spécifié. L'intérêt d'utiliser cette option est le même que pour –dhcp-hostsfile : le fichier spécifié sera rechargé à la réception par dnsmasq d'un signal SIGHUP. Notez qu'il est possible d'encoder l'information via –dhcp-boot en utilisant les noms optionnels bootfile-name, server-ip-address et tftp-server. Ceci permet d'inclure ces options dans un fichier “dhcp-optsfile”.DNSMASQ_SUPPLIED_HOSTNAME

read-ethers

Lis les informations d'hôtes DHCP dans le fichier /etc/ethers. Le format de /etc/ethers est une adresse matérielle suivie, soit par un nom d'hôte, soit par une adresse IP sous la forme de 4 chiffres séparés par des points. Lorsque lu par Dnsmasq, ces lignes ont exactement le même effet que l'option –dhcp-host contenant les mêmes informations. /etc/ethers est relu à la réception d'un signal SIGHUP par Dnsmasq.

dhcp-option=[tag:<label>,[tag:<label>]][encap:<option>,][vi-encap:<entreprise>,][vendor:[<classe_vendeur>],][<option>|option:<nom d'option>],[<valeur>[,<valeur>]]

Spécifie des options différentes ou supplémentaires pour des clients DHCP. Par défaut, Dnsmasq envoie un ensemble standard d'options aux clients DHCP : le masque de réseau et l'adresse de broadcast sont les mêmes que pour l'hôte sur lequel tourne Dnsmasq, et le serveur DNS ainsi que la route par défaut prennent comme valeur l'adresse de la machine sur laquelle tourne Dnsmasq. Si une option de nom de domaine a été définie, son contenu est transmis. Cette option de configuration permet de changer toutes ces valeurs par défaut, ou de spécifier d'autres options. L'option DHCP à transmettre peut être fournie sous forme d'un nombre décimal ou sous la forme “option:<nom d'option>”. Les nombres correspondants aux options sont définis dans la RFC2132 et suivants. Les noms d'options connus par Dnsmasq peuvent être obtenus via “Dnsmasq –help dhcp”. Par exemple, pour définir la route par défaut à 192.168.4.4, il est possible de faire –dhcp-option=3,192.168.4.4 ou –dhcp-option = option:router, 192.168.4.4 ou encore, pour positionner l'adresse du serveur de temps à 192.168.0.4, on peut faire –dhcp-option = 42,192.168.0.4 ou –dhcp-option = option:ntp-server, 192.168.0.4 L'adresse 0.0.0.0 prends ici le sens “d'adresse de la machine sur laquelle tourne Dnsmasq”. Les types de données autorisées sont des adresses IP sous la forme de 4 chiffres séparés par des points, un nombre décimal, une liste de caractères hexadécimaux séparés par des 2 points, ou une chaîne de caractères. Si des labels optionnels sont fournis, alors cette option n'est envoyée qu'aux réseaux dont tous les labels coïncident avec ceux de la requête.
Un traitement spécial est effectué sur les chaînes de caractères fournies pour l'option 119, conformément à la RFC 3397. Les chaînes de caractères ou les adresses IP sous forme de 4 chiffres séparés par des points donnés en arguments de l'option 120 sont traités conformément à la RFC 3361. Les adresses IP sous forme de 4 chiffres séparés par des points suivies par une barre montante “/”, puis une taille de masque sont encodés conformément à la RFC 3442.
Attention : aucun test n'étant fait pour vérifier que des données d'un type adéquat sont envoyées pour un numéro d'option donné, il est tout à fait possible de persuader Dnsmasq de générer des paquets DHCP illégaux par une utilisation incorrecte de cette option. Lorsque la valeur est un nombre décimal, Dnsmasq doit déterminer la taille des données. Cela est fait en examinant le numéro de l'option et/ou la valeur, mais peut-être évité en rajoutant un suffixe d'une lettre comme suit : b = un octet, s = 2 octets, i = 4 octets. Cela sert essentiellement pour des options encapsulées de classes de vendeurs (voir plus bas), pour lesquelles Dnsmasq ne peut déterminer la taille de la valeur. Les données d'options consistant uniquement de points et de décimaux sont interprétées par Dnsmasq comme des adresses IP, et envoyées comme telles. Pour forcer l'envoi sous forme de chaîne de caractère, il est nécessaire d'utiliser des guillemets doubles. Par exemple, l'utilisation de l'option 66 pour fournir une adresse IP sous la forme d'une chaîne de caractères comme nom de serveur TFTP, il est nécessaire de faire comme suit : –dhcp-option=66,1.2.3.4
Les options encapsulées de classes de vendeurs peuvent-être aussi spécifiées en utilisant –dhcp-option : par exemple –dhcp-option=vendor:PXEClient,1,0.0.0.0 envoie l'option encapsulée de classe de vendeur “mftp-address=0.0.0.0” à n'importe quel client dont la classe de vendeur correspond à “PXEClient”. La correspondance pour les classes de vendeur s'effectue sur des sous-chaînes de caractères (voir –dhcp-vendorclass pour plus de détails). Si une option de classe de vendeur (numéro 60) est envoyée par Dnsmasq, alors cela est utilisé pour sélectionner les options encapsulées, de préférence à toute option envoyée par le client. Il est possible d'omettre complètement une classe de vendeur : –dhcp-option=vendor:,1,0.0.0.0 Dans ce cas l'option encapsulée est toujours envoyée.
Les options peuvent-être encapsulées au sein d'autres options : par exemple –dhcp-

option=encap:175, 190, iscsi-client0 enverra l'option 175, au sein de laquelle se trouve l'option 190. Plusieurs options encapsulées avec le même numéro d'option seront correctement combinées au sein d'une seule option encapsulée. Il n'est pas possible de spécifier encap: et vendor: au sein d'une même option dhcp.
La dernière variante pour les options encapsulées est “l'option de Vendeur identifiant le vendeur” (“Vendor-Identifying Vendor Options”) telle que décrite dans le RFC3925. Celles-ci sont spécifiées comme suit : –dhcp-option=vi-encap:2, 10, text Le numéro dans la section vi-encap: est le numéro IANA de l'entreprise servant à identifier cette option. L'adresse 0.0.0.0 n'est pas traitée de manière particulière lorsque fournie dans une option encapsulée.

dhcp-option-force=[tag:<label>,[tag:<label>]][encap:<option>,][vi-encap:<entreprise>,][vendor:[<classe_vendeur>],][<option>|option:<nom d'option>],[<valeur>[,<valeur>]]

Cela fonctionne exactement de la même façon que –dhcp-option sauf que cette option sera toujours envoyée, même si le client ne la demande pas dans la liste de paramêtres requis. Cela est parfois nécessaire, par exemple lors de la fourniture d'options à PXELinux.

dhcp-no-override

Désactive la réutilisation des champs DHCP nom de serveur et nom de fichier comme espace supplémentaire pour les options. Si cela est possible, dnsmasq déplace les informations sur le serveur de démarrage et le nom de fichier (fournis par 'dhcp-boot') en dehors des champs dédiés à cet usage dans les options DHCP. Cet espace supplémentaire est alors disponible dans le paquet DHCP pour d'autres options, mais peut, dans quelques rares cas, perturber des clients vieux ou défectueux. Cette option force le comportement à l'utilisation des valeurs “simples et sûres” afin d'éviter des problèmes dans de tels cas.

dhcp-vendorclass=set:<label>,<classe de vendeur>

Associe une chaîne de classe de vendeur à un label. La plupart des clients DHCP fournissent une “classe de vendeur” (“vendor class”) qui représente, d'une certaine façon, le type d'hôte. Cette option associe des classes de vendeur à des labels, de telle sorte que des options DHCP peuvent-être fournie de manière sélective aux différentes classes d'hôtes. Par exemple, dhcp-vendorclass=set:printers,Hewlett-Packard JetDirect ou dhcp-vendorclass=printers,Hewlett-Packard JetDirect permet de n'allouer des options qu'aux imprimantes HP de la manière suivante : –dhcp-option=tag:printers,3,192.168.4.4 La chaîne de caractères de la classe de vendeur founie en argument est cherchée en temps que sous-chaîne de caractères au sein de la classe de vendeur fournie par le client, de façon à permettre la recherche d'un sous-ensemble de la chaîne de caractères (“fuzzy matching”). Le préfixe set: est optionnel mais autorisé afin de conserver une certaine homogénéité.

dhcp-userclass=set:<label>,<classe utilisateur>

Associe une chaîne de classe d'utilisateur à un label (effectue la recherche sur des sous-chaînes, comme pour les classes de vendeur). La plupart des clients permettent de configurer une “classe d'utilisateur”. Cette option associe une classe d'utilisateur à un label, de telle manière qu'il soit possible de fournir des options DHCP spécifiques à différentes classes d'hôtes. Il est possible, par exemple, d'utiliser ceci pour définir un serveur d'impression différent pour les hôtes de la classe “comptes” et ceux de la classe “ingénierie”.

dhcp-mac=set:<label>,<adresse MAC>

Associe une adresse matérielle (MAC) à un label. L'adresse matérielle peut inclure des jokers. Par exemple –dhcp-mac=set:3com,01:34:23:*:*:* permet de définir le label

“3com” pour n'importe quel hôte dont l'adresse matérielle coïncide avec les critères définis.

dhcp-circuitid=set:<label>,<identifiant de circuit>, –dhcp-remoteid=set:<label>,<identifiant distant>

Associe des options de relais DHCP issus de la RFC3046 à des labels. Cette information peut-être fournie par des relais DHCP. L'identifiant de circuit ou l'identifiant distant est normalement fourni sous la forme d'une chaîne de valeurs hexadécimales séparées par des “:”, mais il est également possible qu'elle le soit sous la forme d'une simple chaîne de caractères. Si l'identifiant de circuit ou d'agent correspond exactement à celui fourni par le relais DHCP, alors le label est apposé.

dhcp-subscrid=set:<label>,<identifiant d'abonné>

Associe des options de relais DHCP issues de la RFC3993 à des labels.

dhcp-proxy[=<adresse ip>].....

Un agent relai DHCP normal est uniquement utilisé pour faire suivre les éléments initiaux de l'interaction avec le serveur DHCP. Une fois que le client est configuré, il communique directement avec le serveur. Cela n'est pas souhaitable si le relais rajoute des informations supplémentaires aux paquets DHCP, telles que celles utilisées dans dhcp-circuitid et dhcp-remoteid. Une implémentation complète de relai peut utiliser l'option serverid-override de la RFC 5107 afin de forcer le serveur DHCP à utiliser le relai en temps que proxy complet, de sorte que tous les paquets passent par le relai. Cette option permet d'obtenir le même résultat pour des relais ne supportant pas la RFC 5107. Fournie seule, elle manipule la valeur de server-id pour toutes les interactions via des relais. Si une liste d'adresses IP est donnée, seules les interactions avec les relais dont l'adresse est dans la liste seront affectées.

dhcp-match=set:<label>,<numéro d'option>|option:<nom d'option>|vi-encap:<entreprise>[,<valeur>]

Si aucune valeur n'est spécifiée, associe le label si le client envoie une option DHCP avec le numéro ou le nom spécifié. Lorsqu'une valeur est fournie, positionne le label seulement dans le cas où l'option est fournie et correspond à la valeur. La valeur peut-être de la forme “01:ff:*:02”, auquel cas le début de l'option doit correspondre (en respectant les jokers). La valeur peut aussi être de la même forme que dans dhcp-option , auquel cas l'option est traitée comme un tableau de valeur, et un des éléments doit correspondre, ainsi

dhcp-match=set:efi-ia32,option:client-arch,6

spécifie le label “efi-ia32” si le numéro 6 apparaît dnas la liste d'architectures envoyé par le client au sein de l'option 93. (se réferer au RFC 4578 pour plus de détails). Si la valeur est un chaine de caractères, celle-ci est recherchée (correspondance en temps que sous-chaîne).

Pour la forme particulière vi-encap:<numéro d'entreprise>, la comparaison se fait avec les classes de vendeur “identifiant de vendeur” (“vendor-identifying vendor classes”) pour l'entreprise dont le numéro est fourni en option. Veuillez vous réferer à la RFC 3925 pour plus de détail.

tag-if=set:<label>[,set:<label>[,tag:<label>[,tag:<label>]]]

Effectue une opération booléenne sur les labels. Si tous les labels apparaissant dans la liste tag:<label> sont positionnés, alors tous les la de la liste “set:<labels>” sont positionnés (ou supprimés, dans le cas où “tag:!<label>” utilisé). Si aucun tag:<label> n'est spécifié, alors tous les labels fournis par set:<label> sont positionnés. N'importe quel nombre de set: ou tag: peuvent être fournis, et l'ordre est sans importance. Les lignes tag-if sont executées dans l'ordre, ce qui fait que si un label dans tag:<label> est un label positionné par une rêgle tag-if, la ligne qui positionne le label doit précéder celle qui le teste.

dhcp-ignore=tag:<label>[,tag:<label>]

Lorsque tous les labels fournis dans l'option sont présents, ignorer l'hôte et ne pas donner de bail DHCP.

dhcp-ignore-names[=tag:<label>[,tag:<label>]]

Lorsque tous les labels fournis dans l'option sont présents, ignorer le nom de machine fourni par l'hôte. Il est à noter que, à la différence de l'option “dhcp-ignore”, il est permis de ne pas fournir de label. Dans ce cas, les noms d'hôtes fournis par les clients DHCP seront toujours ignorés, et les noms d'hôtes seront ajoutés au DNS en utilisant uniquement la configuration dhcp-host de Dnsmasq, ainsi que le contenu des fichiers /etc/hosts et /etc/ethers.

dhcp-generate-names=tag:<label>[,tag:<label>]

Générer un nom pour les clients DHCP qui autrement n'en aurait pas, en utilisant l'adresse MAC sous sa forme hexadécimale, séparée par des tirets. Noter que si un hôte fourni un nom, celui-ci sera utilisé de préférence au nom autogénéré, à moins que –dhcp-ignore-names ne soit positionné.

dhcp-broadcast=[tag:<label>[,tag:<label>]]

Lorsque tous les labels fournis dans l'option sont présents, toujours utiliser le broadcast pour communiquer avec l'hôte lorsque celui-ci n'est pas configuré. Il est possible de ne spécifier aucun label, auquel cas cette option s'applique inconditionnellement. La plupart des clients DHCP nécessitant une réponse par le biais d'un broadcast activent une option dans leur requête, ce qui fait que cela se fait automatiquement, mais ce n'est pas la cas de certains vieux clients BOOTP.

dhcp-boot=[tag:<label>,]<nom de fichier>,[<nom de serveur>[,<adresse de serveur>]]

Spécifie les options BOOTP devant être retournées par le serveur DHCP. Le nom de serveur ainsi que l'adresse sont optionnels : s'ils ne sont pas fournis, le nom est laissé vide et l'adresse fournie est celle de la machine sur laquelle s'exécute Dnsmasq. Si Dnsmasq founit un service TFTP (voir –enable-tftp), alors seul un nom de fichier est requis ici pour permettre un démarrage par le réseau. Si d'éventuels labels sont fournis, ils doivent coïncider avec ceux du client pour que cet élement de configuration lui soit envoyé.

pxe-service=[tag:<label>,]<CSA>,<entrée de menu>[,<nom de fichier>|<type de service de démarrage>][,<adresse de serveur>]

La plupart des ROMS de démarrage PXE ne permettent au système PXE que la simple obtention d'une adresse IP, le téléchargement du fichier spécifié dans dhcp-boot et son exécution. Cependant, le système PXE est capable de fonctions bien plus complexes pour peu que le serveur DHCP soit adapté.

Ceci spécifie l'option de démarrage qui apparaitra dans un menu de démarrage PXE. <CSA> est le type du système client. Seuls des types de services valides apparaitront dans un menu. Les types connus sont x86PC, PC98, IA64_EFI, Alpha, Arc_x86, Intel_Lean_Client, IA32_EFI, BC_EFI, Xscale_EFI et X86-64_EFI; D'autres types peuvent-être spécifiés sous la forme d'une valeur entière. Le paramètre après le texte correspondant à l'entrée dans le menu peut être un nom de fichier, auquel cas Dnsmasq agit comme un serveur de démarrage et indique au client PXE qu'il faut télécharger ce fichier via TFTP, soit depuis ce serveur (l'option enable-tftp doit être spécifiée pour que cela marche), soit depuis un autre serveur TFTP si une adresse de serveur est fournie. Veuillez noter que le suffixe de “couche” (en principe “.0”) est fourni par PXE et ne doit pas être rajouté au nom de fichier. Si une valeur numérique entière est fournir pour le type de démarrage, en remplacement du nom de fichier, le client PXE devra chercher un service de démarrage de ce type sur le réseau. Cette recherche peut être faite via broadcast ou directement auprès d'un serveur si son adresse IP est fournie dans l'option.

Si aucun nom de fichier n'est donné ni aucune valeur de type de service de démarrage n'est fournie (ou qu'une valeur de 0 est donnée pour le type de service), alors l'entrée de menu provoque l'interruption du démarrage par le réseau et la poursuite du démarrage sur un média local.

pxe-prompt=[tag:<label>,]<invite>[,<délai>]

Cette option permet d'afficher une invite à la suite du démarrage PXE. Si un délai est fourni, alors la première entrée du menu de démarrage sera automatiquement exécutée après ce délai. Si le délai vaut 0, alors la première entrée disponible sera exécutée immédiatement. Si pxe-prompt est omis, le système attendra un choix de l'utilisateur s'il existe plusieurs entrées dans le menu, ou démarrera immédiatement dans le cas où il n'y a qu'une seule entrée. Voir pxe-service pour plus de détails sur les entrées de menu. Dnsmasq peut servir de “proxy-DHCP” PXE, dans le cas où un autre serveur DHCP sur le réseau est responsable de l'allocation des adresses IP, auquel cas Dnsmasq se contente de fournir les informations données dans les options pxe-prompt et pxe-service pour permettre le démarrage par le réseau. Ce mode est activé en utilisant le mot-clef proxy dans dhcp-range.

dhcp-lease-max=<nombre>

Limite Dnsmasq à un maximum de <nombre> baux DHCP. Le défaut est de 1000. Cette limite permet d'éviter des attaques de déni de service (“DoS”) par des hôtes créant des milliers de baux et utilisant beaucoup de mémoire dans le processus Dnsmasq.

dhcp-authoritative

Cette option doit être donnée lorsque Dnsmasq est le seul serveur DHCP sur le réseau. Cela change le comportement par défaut qui est celui d'un strict respect des RFC, afin que les requêtes DHCP pour des baux inconnus par des hôtes inconnus ne soient pas ignorées. Cela permet à de nouveaux hôtes d'obtenir des baux sans tenir compte de fastidieuses temporisations (“timeout”). Cela permet également à Dnsmasq de reconstruire sa base de donnée contenant les baux sans que les clients n'aient besoin de redemander un bail, si celle-ci est perdue.

dhcp-alternate-port[=<port serveur>[,<port client>]]

Change les ports utilisés par défaut pour le DHCP. Si cette option est donnée toute seule sans arguments, alors change les ports utilisés pour le DHCP de 67 et 68 respectivement à 1067 et 1068. Si un seul argument est donné, ce numéro est utilisé pour le port serveur et ce numéro plus 1 est utilisé pour le port client. Enfin, en fournissant deux numéros de ports, il est possible de spécifier arbitrairement 2 ports à la fois pour le serveur et pour le client DHCP.

bootp-dynamic[=<identifiant de réseau>[,<identifiant de réseau>]]

Permet l'allocation dynamique d'adresses IP à des clients BOOTP. Utiliser cette option avec précaution, une adresse allouée à un client BOOTP étant perpétuelle, et de fait n'est plus disponibles pour d'autres hôtes. Si aucun argument n'est donné, alors cette option permet une allocation dynamique dans tous les cas. Si des arguments sont spécifiés, alors l'allocation ne se fait que lorsque tous les identifiants coïncident. Il est possible de répeter cette option avec plusieurs jeux d'arguments.

no-ping

Par défaut, le serveur DHCP tente de s'assurer qu'une adresse n'est pas utilisée avant de l'allouer à un hôte. Cela est fait en envoyant une requête ICMP de type “echo request” (aussi connue sous le nom de “ping”) à l'adresse en question. Si le serveur obtient une réponse, alors l'adresse doit déjà être utilisée et une autre est essayée. Cette option permet de supprimer cette vérification. A utiliser avec précaution.

log-dhcp

Traces additionnelles pour le service DHCP : enregistre toutes les options envoyées aux clients DHCP et les labels utilisés pour la détermination de celles-ci.

dhcp-leasefile=<chemin de fichier>

Utilise le fichier dont le chemin est fourni pour stocker les informations de baux DHCP.

dhcp-script=<chemin de fichier>

Lorsqu'un bail DHCP est créé, ou qu'un ancien est supprimé, le fichier dont le chemin est spécifié est exécuté. Le <chemin de fichier> doit être un chemin absolu, aucune recherche n'est effectuée via la variable d'environnement PATH. Les arguments fournis à celui-ci sont soit “add” (“ajouter”), “old” (“ancien”) ou “del” (“supprimer”), suivi de l'adresse MAC de l'hôte puis l'adresse IP et le nom d'hôte si celui-ci est connu.“add” signifie qu'un bail a été créé, “del” signifie qu'il a été supprimé, “old” notifie que le bail existait au lancement de Dnsmasq, ou un changement d'adresse MAC ou de nom d'hôte pour un bail existant (ou, dans le cas où leasefile-ro est spécifié, un changement de durée de bail ou d'identifiant d'hôte). Si l'adresse Mac est d'un type de réseau autre qu'ethernet, il est nécessaire de la préceder du type de réseau, par exemple “06-01:23:45:67:89:ab” pour du token ring. Le processus est exécuté en temps que super-utilisateur (si Dnsmasq a été lancé en temps que “root”), même si Dnsmasq est configuré pour changer son UID pour celle d'un utilisateur non-privilégié.
L'environnement est hérité de celui de l'invocation du processus Dnsmasq, auquel se rajoute quelques unes ou toutes les variables décrites ci-dessous :
DNSMASQ_CLIENT_ID, si l'hôte a fourni un identifiant de client.
DNSMASQ_DOMAIN si le nom de domaine pleinement qualifié de l'hôte est connu, la part relative au domaine y est stockée.
Si le client fournit une information de classe de vendeur, un nom d'hôte, ou des classes d'utilisateur, celles-ci sont fournies dans les variables DNSMASQ_VENDOR_CLASS et DNSMASQ_USER_CLASS0 à DNSMASQ_USER_CLASSn et DNSMASQ_SUPPLIED_HOSTNAME respectivement, mais seulement pour les actions “add” et “old” lorsqu'un hôte reprend un bail existant, ces variables n'étant pas stockées dans la base de baux de Dnsmasq.
Si Dnsmasq a été compilé avec l'option HAVE_BROKEN_RTC (“horloge RTC défectueuse”), alors la durée du bail (en secondes) est stockée dans la variable DNSMASQ_LEASE_LENGTH, sinon la date d'expiration du bail est toujours stocké dans la variable d'environnement DNSMASQ_LEASE_EXPIRES. Le nombre de secondes avant expiration est toujours stocké dans DNSMASQ_TIME_REMAINING.
Si un bail était associé à un nom d'hôte et que celui-ci est supprimé, un évènement de type “old” est généré avec le nouveau statut du bail, c-à-d sans nom d'hôte, et le nom initial est fourni dans la variable d'environnement DNSMASQ_OLD_HOSTNAME.
La variable DNSMASQ_INTERFACE contient le nom de l'interface sur laquelle la requête est arrivée; ceci n'est pas renseigné dans le cas des actions “old” ayant lieu après un redémarrage de dnsmasq.
La variable DNSMASQ_RELAY_ADDRESS est renseignée si le client a utilisé un relai DHCP pour contacter Dnsmasq, si l'adresse IP du relai est connue.
DNSMASQ_TAGS contient tous les labels fournis pendant la transaction DHCP, séparés par des espaces.
Tous les descripteurs de fichiers sont fermés, sauf stdin, stdout et stderr qui sont ouverts sur /dev/null (sauf en mode déverminage).
Le script n'est pas lancé de manière concurrente : au plus une instance du script est executée à la fois (dnsmasq attends qu'une instance de script se termine avant de lancer la suivante). Les changements dans la base des baux nécessitant le lancement du script sont placé en attente dans une queue jusqu'à terminaison d'une instance du script en cours. Si cette mise en queue fait que plusieurs changements d'états apparaissent pour un bail donné avant que le script puisse être lancé, alors les états les plus anciens sont supprimés et lorsque le script sera finalement lancé, ce sera avec l'état courant du bail.

Au démarrage de Dnsmasq, le script sera invoqué pour chacun des baux existants dans le fichier des baux. Le script sera lancé avec l'action “del” pour les baux expirés, et “old” pour les autres. Lorsque Dnsmasq reçoit un signal HUP, le script sera invoqué avec une action “old” pour tous les baux existants.

dhcp-scriptuser

Spécifie l'utilisateur sous lequel le script lease-change doit être exécuté. La valeur par défaut correspond à l'utilisateur root mais peut-être changée par le biais de cette option.

leasefile-ro

Supprimer complètement l'usage du fichier servant de base de donnée pour les baux DHCP. Le fichier ne sera ni créé, ni lu, ni écrit. Change la façon dont le script de changement d'état de bail est lancé (si celui-ci est fourni par le biais de l'option –dhcp-script), de sorte que la base de données de baux puisse être complètement gérée par le script sur un stockage externe. En addition aux actions décrites dans –dhcp-script, le script de changement d'état de bail est appellé une fois, au lancement de Dnsmasq, avec pour seul argument “init”. Lorsqu'appellé de la sorte, le script doit fournir l'état de la base de baux, dans le format de fichier de baux de Dnsmasq, sur sa sortie standard (stdout) et retourner un code de retour de 0. Positionner cette option provoque également une invocation du script de changement d'état de bail à chaque changement de l'identifiant de client, de longueur de bail ou de date d'expiration.

bridge-interface=<interface>,<alias>[,<alias>]

Traiter les requêtes DHCP arrivant sur n'importe laquelle des interfaces <alias> comme si elles arrivaient de l'interface <interface>. Cette option est nécessaire lors de l'utilisation de pont ethernet “ancien mode” sur plate-forme BSD, puisque dans ce cas les paquets arrivent sur des interfaces “tap” n'ont pas d'adresse IP.

domain=<domaine>[,<gamme d'adresses>]

Spécifie le domaine du serveur DHCP. Le domaine peut être donné de manière inconditionnelle (sans spécifier de gamme d'adresses IP) ou pour des gammes d'adresses IP limitées. Cela a deux effets; tout d'abord, le serveur DHCP retourne le domaine à tous les hôtes le demandant, deuxièmement, cela spécifie le domaine valide pour les hôtes DHCP configurés. Le but de cela est de contraindre les noms d'hôte afin qu'aucun hôte sur le LAN ne puisse fournir via DHCP un nom tel que par exemple “microsoft.com” et capturer du trafic de manière illégitime. Si aucun nom de domaine n'est spécifié, alors les noms d'hôtes avec un nom de domaine (c-à-d un point dans le nom) seront interdits et enregistrés dans le journal (logs). Si un suffixe est fourni, alors les noms d'hôtes possédant un domaine sont autorisés, pour peu que le nom de domaine coïncide avec le nom fourni. De plus, si un suffixe est fourni, alors les noms d'hôtes ne possédant pas de nom de domain se voient rajouter le suffixe fourni dans l'option –domain. Ainsi, sur mon réseau, je peux configurer –domain=thekelleys.org.uk et avoir une machine dont le nom DHCP serait “laptop”. L'adresse IP de cette machine sera disponible à la fois pour “laptop” et “laptop.thekelleys.org.uk”. Si la valeur fournie pour <domaine> est “#”, alors le nom de domaine est positionné à la première valeur de la directive “search” du fichier /etc/resolv.conf (ou équivalent). La gamme d'adresses peut être de la forme <adresse ip>,<adresse ip> ou <adresse ip>/<masque de réseau> voire une simple <adresse ip>. Voir –dhcp-fqdn qui peut changer le comportement de dnsmasq relatif aux domaines.

dhcp-fqdn

Dans le mode par défaut, dnsmasq insère les noms non-qualifiés des clients DHCP dans le DNS. Pour cette raison, les noms doivent être uniques, même si deux clients ayant le même nom sont dans deux domaines différents. Si un deuxième client DHCP apparaît ayant le même nom qu'un client déjà existant, ce nom est transféré au nouveau client. Si –dhcp-fqdn est spécifié, ce comportement change : les noms non qualifiés ne sont plus rajoutés dans le DNS, seuls les noms qualifiés le sont. Deux clients DHCP avec le même

nom peuvent tous les deux garder le nom, pour peu que la partie relative au domaine soit différente (c-à-d que les noms pleinements qualifiés diffèrent). Pour d'assurer que tous les noms ont une partie domaine, il doit-y avoir au moins un –domain sans gamme d'adresses de spécifié lorsque l'option –dhcp-fqdn est configurée.

enable-tftp[=<interface>]
: Active la fonction serveur TFTP. Celui-ci est de manière délibérée limité aux fonctions nécessaires au démarrage par le réseau (“net-boot”) d'un client. Seul un accès en lecture est possible; les extensions tsize et blksize sont supportées (tsize est seulement supporté en mode octet). Voir dans la section NOTES les informations relatives à la spécification de l'interface.

tftp-root=<répertoire>[,<interface>]
: Les fichiers à fournir dans les transferts TFTP seront cherchés en prenant le répertoire fourni comme racine. Lorsque cela est fourni, les chemins TFTP incluant “..” sont rejetés, afin d'éviter que les clients ne puissent sortir de la racine spécifiée. Les chemins absolus (commençant par “/”) sont autorisés, mais ils doivent être à la racine TFTP fournie. Si l'option interface est spécifiée, le répertoire n'est utilisé que pour les requêtes TFTP reçues sur cette interface.

tftp-unique-root
: Ajouter l'adresse IP du client TFTP en temps qu'élément de chemin, à la suite de la racine tftp (adresse sous forme de 4 chiffres séparés par des points). Uniquement valable si une racine TFTP est spécifiée et si le répertoire correspond existe. Ainsi, si la valeur pour tftp-root est “/tftp” et que le client d'adresse IP 1.2.3.4 requiert le fichier “monfichier”, alors le chemin effective résultant sera “/tftp/1.2.3.4/monfichier” si /tftp/1.2.3.4 existe, ou “/tftp/monfichier” dans le cas contraire.

tftp-secure
: Active le mode TFTP sécurisé : sans cela, tout fichier lisible par Dnsmasq est disponible via TFTP (les règles de contrôle d'accès unix habituelles s'appliquent). Lorsque l'option –tftp-secure est spécifiée, seuls les fichiers possédés par l'utilisateur sous lequel tourne le processus Dnsmasq sont accessibles. Si Dnsmasq est exécuté en temps que super-utilisateur (“root”), des règles différentes s'appliquent : –tftp-secure n'a aucun effet, mais seuls les fichiers ayant un droit de lecture pour tout le monde sont accessibles. Il n'est pas recommandé d'exécuter Dnsmasq sous l'utilisateur “root” lorsque le service TFTP est activé, et il est formellement déconseillé de le faire sans fournir l'option –tftp-root. Sans cela, en effet, l'accès de tous les fichiers du serveur pour lequel le droit de lecture pour tout le monde est positionné (“world-readable”) devient possible par n'importe quel hôte sur le réseau.

tftp-max=<connexions>
: Définit le nombre maximum de connexions TFTP simultanées autorisées. La valeur par défaut est de 50. Lorsqu'un grand nombre de connexions TFTP est spécifié, il se peut que la limite de nombre de descripteurs de fichiers par processus soit atteinte. Dnsmasq nécessite quelques descripteurs de fichiers, ainsi qu'un descripteur de fichier pour chaque connexion TFTP simultanée et pour chacun des fichiers devant être fournis. De fait, servir le même fichier à n clients ne nécessitera qu'environ n + 10 descripteurs de fichiers, alors que fournir des fichiers tous différents à n clients utilisera environ (2*n) + 10 descripteurs. Si elle est donnée, l'option –tftp-port-range peut affecter le nombre maximum de connexions concurrentes.

tftp-no-blocksize
: Empêche le serveur TFTP de négocier l'option “blocksize” (taille de bloc) avec les clients. Certains clients buggés spécifient cette option mais se comportent ensuite de manière incorrecte si celle-ci est accordée.

tftp-port-range=<début>,<fin>

Un serveur TFTP écoute sur le port prédéfini 69 (“well-known port”) pour l'initiation de la connexion, mais utilise également un port dynamiquement alloué pour chaque connexion. Normalement, ces ports sont alloués par le système d'exploitation, mais cette option permet de spécifier une gamme de ports à utiliser pour les transferts TFTP. Cela peut-être utile si TFTP doit traverser un dispositif garde-barrière (“firewall”). La valeur de début pour la plage de port ne peut-être inférieure à 1025 sauf si dnsmasq tourne en temps que super-utilisateur (“root”). Le nombre de connexions TFTP concurrentes est limitée par la taille de la gamme de ports ainsi spécifiée.

tftp-port-range=<début>,<fin>

Un serveur TFTP écoute sur un numéro de port bien connu (69) pour l'initiation de la connexion, et alloue dynamiquement un port pour chaque connexion. Ces numéros de ports sont en principe alloués par le système d'exploitation, mais cette option permet de spécifier une gamme de ports à utiliser pour les transferts TFTP. Cela peut-être utile lorsque ceux-ci doivent traverser un dispositif garde-barrière (“firewall”). Le début de la plage ne peut-être inférieur à 1024 à moins que Dnsmasq ne fonctionne en temps que super-utilisateur (“root”). Le nombre maximal de connexions TFTP concurrentes est limitée par la taille de la plage de ports ainsi définie.

conf-file=<fichier>

Spécifie un fichier de configuration différent. L'option “conf-file” est également autorisée dans des fichiers de configuration, ce qui permet l'inclusion de multiples fichiers de configuration.

conf-dir=<répertoire>[,<extension de fichier>...]

Lis tous les fichiers du répertoire spécifié et les traite comme des fichiers de configuration. Si des extensions sont données, tout fichier finissant par ces extensions seront ignorés. Tout fichier dont le nom se termine en ~ ou commence par ., ainsi que ceux commençant ou se terminant par # seront systématiquement ignorés. Cette option peut être donnée en ligne de commande ou dans un fichier de configuration.