

Logiciel

DNSmasq : un serveur DNS local pour un PC sous Ubuntu

Sur un réseau, les appareils n'échangent qu'avec des adresses IP.

Les serveurs DNS convertissent les noms d'hôte en adresses IP :

- si le serveur connaît l'adresse IP correspondante, il retourne directement l'adresse IP
- sinon, il demande à son serveur DNS et renvoie la réponse
- si son serveur DNS n'a pas non plus la réponse, il demandera au serveur suivant, etc ...

Un serveur DNS :

- accélère la navigation grâce au cache,
- ne transmet pas les conversions DNS à des serveurs publics (comme google, etc.)
- augmente la sécurité : votre serveur DNS reste en fonction si internet ne fonctionne plus.
- permet des enregistrements personnalisés : avec un serveur DNS à la maison, vous pouvez créer vos propres enregistrements dans le serveur DNS. Par exemple, vous pouvez rediriger framboise.me.local vers un Raspberry Pi

Dnsmasq assure à la fois les fonctions de cache et de serveur DHCP. Il gère les baux DHCP statiques et dynamiques.

Dans cette page, nous installons **dnsmasq** sur une machine pour lui donner accès aux machines du réseau local.

Voir aussi :

- [DnsMasq : utiliser votre Raspberry Pi comme serveur DNS \(et accélérer Internet\)](#)
- [Utilisation du plug-in DNSMasq de NetworkManager](#)
- [dnsmasq : depuis Ubuntu 18.04](#)
- [DNSMasq : Utilisation du plug-in DNSMasq de NetworkManager](#)

Pré-requis

Installation

?? Cohabitation avec systemd : si on installe dnsmasq comme serveur DNS pour un réseau local, dnsmasq écoute sur le port 53 qui est déjà utilisé par systemd-resolved.

!!! Nous présentons ici la méthode pour faire fonctionner les deux services ensemble.

Voir [dnsmasq : depuis Ubuntu 18.04](#)

1. **Installez le paquet dnsmasq** ou en ligne de commande :

```
...@...:~$ sudo apt install dnsmasq
```

Configuration

Voir :

- [Comment configurer Dnsmasq, serveur dns et dhcp léger](#)
- [Dnsmasq.conf : options](#)

Comme précisé dans **/etc/default/dnsmasq**,

tous les fichiers du répertoire **/etc/dnsmasq.d** sont lus par dnsmasq en tant que fichiers de configuration,

sauf si leurs noms se terminent par ".dpkg-dist", ". Dpkg-old" ou ".dpkg-new".

On ne touche donc pas à :

- **/etc/resolv.conf** qui renvoie vers **127.0.0.1** (dnsmasq installé sur cette machine)
 - ⇒ remplacé par **/etc/resolv.dnsmasq**
- **/etc/dnsmasq.conf** (fichier de configuration principal, entièrement en commentaire)
 - ⇒ remplacé par **/etc/dnsmasq.d/local.conf**



Les fichiers de configuration à utiliser sont :

Fichier à utiliser		au lieu de
/etc/hosts	résolution des noms d'hôte du réseau	
/etc/resolv.dnsmasq	serveurs DNS à utiliser	/etc/resolv.conf
/etc/dnsmasq.d/*.conf	fichier de configuration dans lequel une ligne demande d'utiliser /etc/resolv.dnsmasq	/etc/dnsmasq.conf

1. **Créez le répertoire /etc/systemd/resolved.conf.d** pour les configurations de systemd-resolved (on ne touchera pas à /etc/systemd/resolved.conf, car il peut être écrasé lors de la mise à niveau) :

```
...@...:~$ sudo mkdir /etc/systemd/resolved.conf.d
```

2. **Désactivez l'écoute sur le port 53** pour systemd-resolved en créant avec les droits d'administration le fichier **/etc/systemd/resolved.conf.d/noresolved.conf** :

[/etc/systemd/resolved.conf.d/noresolved.conf](#)

```
[Resolve]
DNSStubListener=no
```

3. **Redémarrez systemd-resolved** :

```
...@...:~$ sudo systemctl restart systemd-resolved.service
```

4. **Supprimez et recréez /etc/resolv.conf** ¹⁾ :

```
...@...:~$ sudo rm /etc/resolv.conf
...@...:~$ sudo touch /etc/resolv.conf
```

5. **Empêchez l'écrasement de /etc/resolv.conf par NetworkManager** en créant avec les droits d'administration le fichier **/etc/NetworkManager/conf.d/disable-resolv.conf** :

[/etc/NetworkManager/conf.d/disable-resolv.conf](#)

```
[main]
dns=none
```

6. **Redémarrez NetworkManager** :

```
...@...:~$ sudo systemctl restart NetworkManager.service
```

7. **Empêchez dnsmasq d'utiliser resolv.conf de NetworkManager** en créant avec les droits d'administration le fichier **/etc/dnsmasq.d/nmresolv.conf** :

[/etc/dnsmasq.d/nmresolv.conf](#)

```
resolv-file=/var/run/NetworkManager/resolv.conf
```

Fichier /etc/dnsmasq.d/*.conf (pour ne pas toucher à /etc/dnsmasq.conf)

user et group

utilisateur et groupe sous lequel dnsmasq sera lancé

; addn-hosts

fichier de configuration différent que /etc/hosts pour la résolution des noms

; log-queries

Journaliser les requêtes DNS (dans /var/log/messages par défaut), utile pour du debug

domain-needed

Ignore les requêtes sans nom de domaine complet.

Par exemple, «machine» ne sera pas transmis aux serveurs DNS de votre FAI, alors que «machine.domain.com» le sera.

; bogus-priv

Pas de résolution inverse pour les réseaux privés.

Les requêtes pour des adresses IP privées (ie 192.168.x.x, etc...) qui ne sont pas traitées par /etc/hosts ou le fichier de baux DHCP

retournent «no such domain» au lieu d'être transmises aux serveurs amont.



; filterwin2k

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 soulignés (requêtes pour des serveurs LDAP).

; domain=maison.lan

Domaine ajouté aux noms sans domaine pour expand-hosts.

Attribue également un domaine aux clients DHCP.


; expand-hosts

Ajoute le nom de domaine ci-dessus aux noms simples (ne contenant pas de point dans le nom) contenus dans le fichier `/etc/hosts`, de la même façon que pour le service DHCP.

</WRAP>

1. Configurez le DNS en créant avec les droits d'administration le fichier **`/etc/dnsmasq.d/network-manager.conf`** :

[`/etc/dnsmasq.d/network-manager.conf`](#)



```
# Ignore les requêtes sans nom
de domaine complet
domain-needed
# Pas de résolution inverse
pour les réseaux privés
bogus-priv

# Fichier définissant les
serveurs (au lieu de
/etc/resolv.conf)
resolv-file=/etc/resolv.dnsmasq
# Essayer les serveurs dans
l'ordre du fichier
/etc/resolv.dnsmasq
strict-order
```

2. **Configurez dnsmasq** en créant avec les droits d'administration le fichier **`/etc/dnsmasq.d/mondns.conf`**, par exemple :


[`/etc/dnsmasq.d/mondomaine.conf`](#)

```
address=/pc1.mondomaine/192.168
.0.1
address=/framboise.mondomaine/1
92.168.0.31
address=/framboise4.mondomaine/
192.168.0.32
```

Autre exemple :

- resolv :

[`/etc/dnsmasq.d/resolv.conf`](#)



```
# Teste les changements du
fichier resolv.dnsmasq et
le relit
no-poll
```

- server :

[/etc/dnsmasq.d/server.conf](#)

```
# autres serveurs de noms,
avec des spécifications de
domaine si ce sont des
domaines non publics :
server=/localnet/192.168.0
.1
# Exemple de routage des
requêtes PTR vers les
serveurs de noms
# envoie toutes les
requêtes adresse->nom pour
192.168.3/24 au serveur de
noms 10.1.2.3
server=/3.168.192.in-
addr.arpa/10.1.2.3
```

- local :

[/etc/dnsmasq.d/local.conf](#)

```
# domaines uniquement
locaux
# Les requêtes pour ces
domaines ne sont traitées
qu'à partir de /etc/hosts
ou DHCP
local=/localnet/
```

- address :

[/etc/dnsmasq.d/address.conf](#)

```
# forcer une adresse IP
pour ces domaines.
# Exemple : renvoyer
n'importe quel hôte de
```

```
doubleclick.net vers le
serveur Web local :
address=/doubleclick.net/1
27.0.0.1
```

- interface :

</etc/dnsmasq.d/interface.conf>

```
interface=eth0
```

- hosts :

</etc/dnsmasq.d/hosts.conf>

```
# Pour que dnsmasq ne lise
pas /etc/hosts
no-hosts
```



- dhcp :

</etc/dnsmasq.d/dhcp.conf>

```
# Activer le serveur DHCP
intégr  avec la plage
d'adresses disponibles et
 ventuellement une dur e
de bail :
dhcp-
range=192.168.0.100,192.16
8.0.150,24h
# Baux statiques (adresse
MAC,nom,adresse IP, dur e
du bail)
#   Baux permanents
dhcp-
host=00:24:d4:af:a8:0c,fre
ebox-
server,192.168.0.254,infin
ite
dhcp-
host=00:24:d4:7c:59:53,fre
ebox-
player,192.168.0.253,infin
ite
```



```
dhcp-
host=2C:B0:5D:8B:6C:12,sw
etgear,192.168.0.239,infin
ite
dhcp-
host=14:DA:E9:6B:B2:03,omv
,192.168.0.250,infinite
dhcp-
host=00:50:43:01:51:9A,she
ebian,192.168.0.252,infini
te
dhcp-
host=00:22:F4:42:A3:B1,pic
untu,192.168.0.204,infini
e
# Baux à durée limitée
dhcp-
host=d0:66:7b:03:fa:66,sam
sungtv,192.168.0.20,48h
dhcp-
host=00:1e:8f:61:a4:0a,imp
rimante,192.168.0.21,48h
dhcp-
host=E0:2A:82:5B:34:F8,dm
adix,192.168.0.10,48h
# Donner l'adresse IP
192.168.0.115 et un bail
permanent à la machine qui
annonce se nommer
SqueezeboxTouch
dhcp-
host=SqueezeboxTouch,192.1
68.0.115,infinite
```

◦ router :

[/etc/dnsmasq.d/router.conf](#)

```
# Remplacer la route par
défaut fournie par Dnsmasq
(qui suppose que le
routeur est sur la même
machine que dnsmasq)
# ici, mettre l'IP de la
freebox comme routeur par
défaut
dhcp-
option=option:router,192.1
68.0.254
```


- cache-size :

[/etc/dnsmasq.d/cache.conf](#)

```
# taille de la mémoire  
cache  
cache-size=256
```

Journalisation

On peut activer indépendamment les traces DNS et DHCP.
Un exemple avec les deux activés (commenter log-queries
ou log-dhcp pour inhiber) :

[/etc/dnsmasq.d/log.conf](#)

```
# Fichier de sortie du journal  
log-facility=/var/log/dnsmasq.log  
# Journaliser les requêtes DNS via  
Dnsmasq  
log-queries  
# Journaliser beaucoup  
d'informations supplémentaires sur  
les transactions DHCP  
log-dhcp
```



Ensuite

1. **redémarrez dnsmasq :**

```
...@...:~$ sudo systemctl restart  
dnsmasq
```

2. Pour **utiliser dnsmasq pour la résolution**, éditez
avec les droits d'administration le fichier
/etc/resolv.conf :

[/etc/resolv.conf](#)

```
# Use local dnsmasq for  
resolving  
nameserver 127.0.0.1
```

Fichier /etc/hosts du serveur DNS





Inutile d'associer freebox-server = 192.168.0.254, cela ne fonctionnera pas, même si une règle dhcp-host est spécifiée dans le fichier de configuration.

La freebox-server est en effet configurée en IP statique sur la freebox (192.168.0.254). Elle ne lance pas de requête DHCP → dnsmasq ne reçoit donc pas de requête DHCP de la part de la freebox.



De même pour le Raspberry Pi qui héberge le serveur dnsmasq et est configuré en IP statique (bail statique dans la box ou fichier /etc/network/interfaces).

Mais en mettant ces adresses dans /etc/hosts, dnsmasq est configuré par défaut pour les lire.

Dans le fichier /etc/hosts, il ne faut pas indiquer **framboise** pour 127.0.0.1 sinon dnsmasq répondra à une requête DNS externe par la réponse framboise → 127.0.0.1, 192.168.0.31. Le PC distant contactera alors 127.0.0.1 (donc lui-même, alors qu'il pensait contacter framboise).

Voici un exemple de contenu du fichier /etc/hosts de framboise :

[/etc/hosts](#)

```
127.0.0.1 localhost
192.168.0.31 framboise
192.168.0.31 dns.local
```

Fichier /etc/resolv.dnsmasq (pour ne pas toucher à /etc/resolv.conf)



Le fichier resolv.conf renvoie vers dnsmasq qui est installé sur cette

machine :

Le fichier `/etc/resolv.dnsmasq` spécifie les serveurs DNS externes à utiliser par dnsmasq :



[/etc/resolv.conf](#)

```
nameserver  
127.0.0.1
```



Dnsmasq a été configuré pour respecter l'ordre de ce fichier (les préférés en premiers), cf. directive `strict-order`

Voici un exemple :

[/etc/resolv.dnsmasq](#)



```
# Free  
nameserver 212.27.40.240  
nameserver 212.27.40.241  
  
# OpenDNS  
nameserver 208.67.222.222  
nameserver 208.67.220.220  
  
# OVH  
nameserver 91.121.161.184  
nameserver 91.121.164.227  
nameserver 188.165.197.144  
  
# Google  
nameserver 8.8.8.8  
nameserver 4.4.4.4
```

Les adresses IP statiques

Solution sur framboise :

Dans **`/etc/hosts`**, indiquer les hôtes ayant une adresse IP statique et configurer dnsmasq pour lire le fichier **`/etc/hosts`** au démarrage. Pour cela, commenter la règle suivante :

#no-hosts



Dans le fichier `/etc/hosts`, il ne faut pas indiquer 'framboise' pour 127.0.0.1 sinon dnsmasq répondra à une requête DNS externe par la réponse framboise → 127.0.0.1, 192.168.0.250. L'hôte distant contactera donc 127.0.0.1 (c'est-à-dire lui-même, alors qu'il pensait contacter framboise)

Contenu de `/etc/hosts` de framboise

On spécifie les hôtes avec une adresse IP statique, donc non allouée pas dnsmasq.

[/etc/hosts](#)

```
127.0.0.1 localhost
192.168.0.250 sheebian
sheebian.maison.lan
```



Contenu de `/etc/resolv.conf` de sheebian

Ce fichier spécifie le serveur DNS à utiliser, donc renvoie vers dnsmasq qui est installé sur cette machine.

[/etc/resolv.conf](#)

```
nameserver 127.0.0.1
```

Contenu de `/etc/resolv.dnsmasq` de framboise

Cf la page officielle de man en français :

<http://www.linuxcertif.com/man/8/dnsmasq/>

recopiée ici : [Paramètres de dnsmasq.conf](#)

Les directives de configuration seront écrites dans un fichier **`/etc/dnsmasq.d/local.conf`** créé pour l'occasion et pris en charge automatiquement ²⁾

Ce fichier spécifie en particulier les DNS externes à utiliser par dnsmasq (lignes **nameserver**). Dnsmasq a été configuré pour respecter l'ordre de ce fichier (les préférés en premiers), cf. directive `strict-order`

[/etc/dnsmasq.d/local.conf](#)

```
# Free
nameserver 212.27.40.240
nameserver 212.27.40.241

# OpenDNS
nameserver 208.67.222.222
nameserver 208.67.220.220

# OVH
nameserver 91.121.161.184
nameserver 91.121.164.227
nameserver 188.165.197.144

# Google
nameserver 8.8.8.8
nameserver 4.4.4.4
```

éditez avec les droits d'administration le fichier **/etc/dnsmasq.d/local.conf** pour paramétrer les adresses IP voulues comme ceci :



On commence par empêcher de faire sortir sur internet les requêtes de domaines locaux :

Pour que dnsmasq ajoute automatiquement le nom de domaine quand il sert la demande, ajoutez l'option **expand-hosts** et la définition du nom de domaine (**domain=**).

Avec **expand-hosts**, la recherche DNS pour **hostname.your_domain.com** sauf si **your_domain.com** est spécifié dans l'option **adresse**. Par exemple,

```
domain=your_domain.com
expand-hosts
address=/zirconium.your_domain.com/zr.your_domain.com/192.168.1.31
```

Pour faire des recherches DNS inverses, utiliser **ptr record** :

```
address=/host.example.net/10.1.2.30
ptr-record=30.2.1.10.in-addr.arpa,"host.example.net"
```

Pour créer des baux statiques, utiliser la méthode **dhcp-host** pour les adresses ci-dessus, mais avec des adresses MAC pour ceux qui en ont besoin, par exemple :

```
# This entry is simply a static DNS address,
great for mapping print servers, etc to names
```

```
dhcp-host=zinc,192.168.1.30
# This entry assigns the given IP address to
the MAC address for static IP addresses
# Note that the IP address listed does NOT
have to be in the DHCP range given, just on
the same subnet
dhcp-
host=11:22:33:44:55:66,zinc,192.168.1.30,infi
nite
```

```
domain-needed
bogus-priv
```

[/etc/dnsmasq.d/local.conf](#)



```
domain-needed
bogus-priv
cache-size=1024

expand-hosts
domain=lan

resolv-file=/etc/resolv.dnsmasq

# Free
nameserver 212.27.40.240
nameserver 212.27.40.241

# OpenDNS
nameserver 208.67.222.222
nameserver 208.67.220.220

# OVH
nameserver 91.121.161.184
nameserver 91.121.164.227
nameserver 188.165.197.144

# Google
nameserver 8.8.8.8
nameserver 4.4.4.4

# Adresses à forcer
address=/localhost/127.0.0.1
address=/framboise/192.168.0.100
address=/chateau/192.168.0.1
address=/trianon/192.168.0.2
address=/framboise.lan/192.168.0.10
0
address=/chateau.lan/192.168.0.1
```

```
address=/trianon.lan/192.168.0.2
```

domain-needed

Interdit à Dnsmasq de transmettre en amont les requêtes de noms simples (sans point ni nom de domaine).

- Si le nom n'est pas dans **/etc/hosts** ou dans la liste des baux DHCP, dnsmasq répond "non trouvé".
- Avec **bogus-priv**, cela évite de faire sortir les requêtes de domaines locaux

; bogus-priv

dnsmasq ne transmet pas aux serveurs DNS amont les requêtes DNS inverses pour des adresses IP privées (ie 192.168.x.x, etc...) qui ne sont ni dans **/etc/hosts** ni dans les baux DHCP.

- Il retourne dans ce cas "no such domain".

; cache-size=<taille>

taille du cache de Dnsmasq

- valeur par défaut : 150 noms.
- une valeur de zéro désactive le cache.

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

domaine du serveur DHCP.





- Ce domaine local sera ajouté aux noms des machines assignées par le DHCP
- Le domaine peut être donné :
 - sans spécifier de gamme d'adresses IP
 - ou pour des gammes d'adresses IP limitées.
- Cela a deux effets :
 1. le serveur DHCP retourne le domaine à tous les hôtes qui le demandent
 2. cela spécifie le domaine valide pour les hôtes DHCP configurés.
- cela empêche un hôte sur le LAN de fournir via DHCP un nom tel que par exemple "microsoft.com" et capturer illégitimement du trafic.
- Si aucun nom de domaine n'est spécifié, les noms d'hôtes avec un nom de domaine (avec un point) seront interdits et enregistrés dans le journal (logs).
- Si un suffixe est fourni,
 - les noms d'hôtes possédant un domaine sont autorisés si le nom de domaine coïncide avec <domaine>
 - les noms d'hôtes ne possédant pas de nom de domaine se voient rajouter le suffixe <domaine>
 - Par exemple, sur mon réseau, je peux configurer **domain=thekelleys.org.uk** avec une machine dont le nom DHCP serait **laptop**.
 - L'adresse IP de cette machine sera disponible à la fois



- pour **laptop**
- et **laptop.thekelley
s.org.uk**

- Si la valeur fournie pour <domaine> est "#", 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 est de la forme **<adresse ip>, <adresse ip> ou <adresse ip>/<masque de réseau>** voire une simple <adresse ip>.

; expand-hosts

Ajoute le nom de domaine <domaine> défini par **domain=<domaine>** aux noms simples (dont le nom ne contient pas de point) :

- contenus dans le fichier **/etc/hosts**
- et pour le service DHCP

; resolv-file=<fichier>

Lit 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 remplace le fichier par défaut, le contenu des suivants est rajouté dans la liste des fichiers à consulter.

- Seul le dernier fichier modifié sera chargé en mémoire.

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

adresse IP d'un serveur de nom amont.

- Cette option n'empêche pas la lecture du fichier **/etc/resolv.conf**.
- Si un ou plusieurs noms de domaine sont fournis,
 - ce serveur ne concernera que ce ou ces domaines : toute requête concernant les domaines <domaine> ne sera adressée qu'à ce serveur.
 - Cette option est destinée aux serveurs de nom privés : si, sur votre réseau, un serveur de nom a pour adresse IP **192.168.1.1** et résout les noms de la forme xxx.internal.thekelleys.org.uk,
 - **server/internal.thekelleys.org.uk/192.168.1.1** enverra les requêtes pour les machines internes vers ce serveur de nom,
 - toutes les autres requêtes seront adressées aux serveurs indiqués



dans le fichier
/etc/resolv.conf.

- Si le domaine spécifié est vide (/), ce serveur ne concerne que les noms "non qualifiés", c'est-à-dire les noms ne possédant pas de point.
- On peut préciser un port non standard à la suite des adresses IP en utilisant le caractère #.
- On peut mettre plus d'une option server 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** → **1.2.3.4**,
- mais ***www.google.com** → **2.3.4.5**.
- L'adresse spéciale # signifie "utiliser les serveurs standards", ainsi

```
server=/google.co  
m/1.2.3.4  
server=/www.googl  
e.com/#
```

enverra les requêtes
pour ***.google.com** → **1.2.3.4**

- mais ***www.google.com** ira comme d'habitude aux serveurs définis par



défaut.

- On peut aussi donner un nom de domaine mais sans adresse IP. C'est alors un domaine local : dnsmasq doit répondre aux requêtes le concernant à partir des entrées du fichier **/etc/hosts** ou des baux DHCP, et ne jamais transmettre les requêtes aux serveurs amont
- **local** est synonyme de **server** pour clarifier l'utilisation de cette option pour cet usage particulier.
- La chaîne de caractères optionnelle suivant le caractère @ définit la source que Dnsmasq doit utiliser pour les réponses à ce serveur de nom.
 - Ce doit être une adresses IP appartenant à la machine sur laquelle tourne Dnsmasq ; sinon la ligne sera ignorée et une erreur sera consignée dans le journal des événements.
 - 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 dont l'adresse source est spécifiée, mais il est possible de la donner directement dans la spécification de l'adresse source.

; address=/nom_machine/adresse_ip

dhcp-host=nom_machine,adresse_ip

définit une adresse IP
pour la machine
nom_machine ; on peut
mettre plusieurs lignes.
exemples :

```
address=/zinc/192
.168.1.30
address=/zirconiu
m/zr/192.168.1.31
dhcp-
host=zinc,192.168
.1.30
```

autres exemples :

```
address=/localhos
t/127.0.0.1 # le
localhost de la
machine depuis
laquelle on
consulte le
serveur
address=/frambois
e/192.168.0.100 #
domaine framboise
et ses sous-
domaines
*.framboise
address=/chateau/
192.168.0.1 #
domaine chateau
et ses sous-
domaines
*.chateau
address=/trianon/
192.168.0.2 #:
domaine trianon
et ses sous-
domaines
*.trianon
address=/frambois
e.lan/192.168.0.1
00 # domaine
framboise.lan et
ses sous-domaines
*.framboise.lan
```



etc. :

</WRAP>

Pour renseigner ce fichier, voir
[Paramètres de dnsmasq.conf](#)

C'est tout. Redémarrer le service en
tapant la commande :

```
$ sudo service dnsmasq  
restart
```

Désormais, les domaines
framboise.lan, etc ainsi que leurs
sous-domaines (*.framboise.lan,
etc.) existent et permettent
l'utilisation des sous-domaines
automatiques.

Il ne reste qu'à déclarer les serveurs
DNS. Éditez avec les droits
d'administration le fichier
/etc/resolv.dnsmasq pour y écrire
l'adresse IP des serveurs DNS
comme ceci :

[/etc/resolv.dnsmasq](#)

```
nameserver  
192.168.0.100  
nameserver  
192.168.0.254
```



Ne pas oublier de
laisser l'adresse de la
box (ici,
192.168.0.254)

Un exemple

Créer un petit intranet «maison» :

- avec un nom de domaine qui
ne sera fonctionnel que sur le
LAN : **mondomaine.lan**



- en IP privées de classe C
- serveur DNS :
 - nom :
ns.mondomaine.lan
 - adresse IP :
192.168.0.100
- Adresses du DNS du fournisseur d'accès :
212.27.32.5 212.27.32.5
- machines :
 - machine1 :
 - nom :
machine1.mondomaine.lan
 - adresse IP :
192.168.0.1
 - machine2 :
 - nom :
machine2.mondomaine.lan
 - adresse IP :
192.168.0.2
 - machineX :
 - nom :
machineX.mondomaine.lan
 - adresse IP :
192.168.0.X

Carte du réseau :

Nom d'hôte	Nom de domaine	nom complet de l'hôte	Adresse IP
serveur DNS	mondomaine.lan	ns.mondomaine.lan	192.168.0.100
machine1	mondomaine.lan	machine1.mondomaine.lan	192.168.0.1
machine2	mondomaine.lan	machine2.mondomaine.lan	192.168.0.2
machineX	mondomaine.lan	machineX.mondomaine.lan	192.168.0.X

Configuration des adresses


Déclarer le serveur DNS d'adresse 192.168.0.1 en premier dans le fichier /etc/resolv.conf en mettant sa ligne au début.

Les adresses de DNS du fournisseur d'accès sont spécifiées dans le fichier /etc/named.conf par l'instruction forwarders.

Fixer l'adresse IP du serveur

éditez avec les droits d'administration le fichier **/etc/network/interfaces** pour le modifier comme ceci :

[/etc/network/interfaces](#)



```
# This file
describes the
network
interfaces
available on your
system
# and how to
activate them.
For more
information, see
interfaces(5).
# The loopback
network interface
auto lo
iface lo inet
loopback

# The primary
network interface
auto eth0
iface eth0 inet
static
address
192.168.1.254
netmask
255.255.255.0
network
192.168.1.0
broadcast
192.168.1.255
gateway
192.168.1.1
# dns-* options
are implemented
by the resolvconf
package, if
installed
dns-nameservers
192.168.1.1
```


Configuration des noms d'hôte

Fichier `/etc/hostname`

éditez avec les droits d'administration le fichier **`/etc/hostname`** pour renseigner le nom du serveur DNS comme ceci :

[/etc/hostname](#)

```
dns.mondomaine.la  
n
```

Démarrer le service :

- `/etc/init.d/hostname.s
h start`

Fichier `/etc/hosts`

Éditez avec les droits d'administration le fichier **`/etc/hosts`** pour y inscrire le nom du serveur et son adresse IP :

[/etc/hosts](#)

```
127.0.0.1  
localhost.localdo  
main  
localhost  
192.168.0.200  
dns.mondomaine.la  
n      dns
```

Fichier `/etc/host.conf` (vérification)

Éditez avec les droits d'administration le fichier **`/etc/host.conf`** pour y insérer les paramètres suivants (normalement c'est déjà fait par l'installation de bind9) :

[/etc/host.conf](#)



```
order hosts, bind
multi on
```

Fichier `/etc/resolv.conf`

Éditez avec les droits d'administration le fichier **`/etc/resolv.conf`** pour y entrer le domaine, la zone de recherche et le nom du serveur DNS :

[`/etc/resolv.conf`](#)

```
domain
mondomaine.lan
search
mondomaine.lan
nameserver
192.168.0.200
```

Utilisation

Vérification



Il faut installer le paquet **`dnsutils`** ou en console :

```
$ sudo apt
install
dnsutils
```

1. Testez le serveur DNS :

- **host** teste la résolution de noms et la recherche inverse :

```
...@...:~$ host
nom_du_serveur
...@...:~$ host
@IP_du_serveur
```

Par exemple :

```
...@...:~$ host -
a.mondomaine.lan
```



```
...@...:~$ host  
test1.mondomaine.  
lan  
...@...:~$ $ host  
test2.mondomaine.  
lan  
...@...:~$ host  
test3.mondomaine.  
lan
```

- **nslookup**³⁾ cherche sur un serveur de noms les informations concernant un domaine ou un hôte et diagnostique les problèmes de configuration du DNS :

```
...@...:~$  
nslookup  
nom_du_serveur  
..@...:~$  
nslookup  
@IP_du_serveur
```

- **dig** permet sensiblement la même chose que « nslookup » :

```
...@...:~$ dig  
nom_du_serveur  
...@...:~$ dig  
nom_de_domaine  
...@...:~$ dig -x  
@IP_du_serveur
```

Exemple :

```
...@...:~$ dig  
z.framboise4.parc  
...;; ANSWER  
SECTION:  
z.framboise4.parc  
. 0 IN A  
192.168.0.32  
...  
;; SERVER:  
127.0.0.1#53(127.  
0.0.1)
```

...

- **ping ::**

```
...@...:~$ ping  
nom_du_serveur  
...@...:~$ ping  
adresse_IP_du_ser  
veur
```

2. pour redonner une @ IP au client :

```
...@...:~$ dhclient  
eth0
```

Pour aller plus loin

Voici quelques définitions utiles à la compréhension et au paramétrage du DNS :



serveur maître

s'occupe des
enregistrements DNS
d'un domaine
enregistré ou non (par
exemple pour le cas
d'un réseau local fermé
comme celui de notre
tuto maison.lan)

; une zone

ensemble des
enregistrements DNS
pour un nom de
domaine. Il existe une
zone de recherche
directe
(correspondance entre
le nom d'hôte et
l'adresse IP) et une
zone de recherche
inverse



(correspondance entre l'adresse IP et le nom d'hôte)

; serveur secondaire

assure la redondance du serveur maître et prend le relais de celui-ci en cas de panne

; serveur esclave

une réplcation du serveur maître utilisé comme sauvegarde

; serveur cache

stocke les requêtes déjà résolues ce qui permet d'économiser la bande passante et de réduire le temps de latence

Maintenant vous disposez d'un serveur DNS digne de ce nom (sans faire de jeu de mots). Le DNS est important surtout si vous souhaitez installer d'autres services comme la messagerie par exemple.

Problèmes rencontrés

- [Comment éviter les conflits entre dnsmasq et systemd-resolved ?](#)
- [DNSMasq : Utilisation du plugin DNSMasq de NetworkManager](#)

Questions et Réponses

=?= depuis Ubuntu 18.04 ===

?? Cohabitation avec systemd : depuis Ubuntu 18.04, dnsmasq ne fait plus de résolution DNS



!!! Solution de contournement :
désactiver le resolver natif
systemd-resolved.service et
utiliser **dnsmasq**.

Création d'un fichier pour **dnsmasq**
pour y renseigner ses serveurs DNS
et les autres commandes de
dnsmasq.

Créez avec les droits
d'administration le fichier
/etc/NetworkManager/dnsmasq.d/monresolv.conf pour y écrire :

[/etc/NetworkManager/dnsmasq.d/monresolv.conf](#)

```
cache-size=1000
except-
interface=lo
server=/localhost
/127.0.0.1
server=/nom.domai
ne.tld/adresse_ip
_serveur_dns_1
server=/nom.domai
ne.tld/adresse_ip
_serveur_dns_2

# adresses :
address=/domaine1
.tld/adresse_ip_1
# domaine
domaine1.tld et
ses sous-domaines
*.domaine1.tld
```

Stoppez et désactivez le service
systemd-resolved :

```
$ sudo systemctl disable
systemd-resolved.service
...
Removed
/etc/systemd/system/multi-
user.target.wants/systemd-
resolved.service.
Removed
/etc/systemd/system/dbus-
```



```
org.freedesktop.resolve1.s  
ervice.  
$ sudo systemctl stop  
systemd-resolved.service
```

Supprimez le fichier resolv.conf :

```
$ sudo rm /etc/resolv.conf
```

Éditez avec les droits
d'administration le fichier
**/etc/NetworkManager/NetworkM
anager.conf** et ajoutez la méthode
dns=dnsmasq :

[/etc/NetworkManager/NetworkMana
ger.conf](#)

```
[main]  
plugins=ifupdown,  
keyfile  
dns=dnsmasq
```

Redémarrez le service
NetworkManager :

```
$ sudo systemctl restart  
NetworkManager
```

Vérifiez que le service dnsmasq est
bien lancé :

```
$ sudo ps -ef | grep  
dnsmasq  
...  
xxxxxxx 13654 4831 0  
09:57 pts/0 00:00:00  
grep --color=auto dnsmasq
```

Vérifiez le fichier resolv.conf :

```
$ cat /etc/resolv.conf  
# Generated by  
NetworkManager  
nameserver 127.0.1.1
```

source :

[https://it.izero.fr/linux-remplacer-res
olver-dns-systemd-resolved-par-
dnsmasq/](https://it.izero.fr/linux-remplacer-resolver-dns-systemd-resolved-par-dnsmasq/)



Désinstallation

Pour supprimer cette application, il suffit de supprimer son paquet.

Selon la méthode choisie, la configuration globale de l'application est conservée ou supprimée.

Les journaux du système, et les fichiers de préférence des utilisateurs dans leurs dossiers personnels sont toujours conservés.

Voir aussi

- **(fr)**
http://irp.nain-t.net/doku.php/160dns:30_construire_un_dns
et le reste du site
- **(fr)**
<http://www.ced-info.com/administration-reseaux/bind9-installation-et-parametrage>
- **(fr)**
<http://linux.crdp.ac-caen.fr/Lcs3/x870.html>
- **(fr)**
<http://webadonf.net/2011/03/configurer-un-serveur-dns-avec-bind9-sur-debian-squeeze/>

Cohabitation avec systemd

- **(en)**
<https://www.osradar.com/how-to-install-configure-dnsmasq-on-ubuntu-18-04-lts/>
- **(en)**
<https://unix.stackexchange.com/questions/304050/how-to-avoid-conflicts-between-dnsmasq-and-systemd-resolved>
- **(en)**
<https://bbs.archlinux.org/viewt>

[opic.php?id=246649](https://askubuntu.com/questions/246649)

- (en) <https://askubuntu.com/questions/1012641/dns-set-to-systemds-127-0-0-53-how-to-change-permanently>
- (en) <https://askubuntu.com/questions/898605/how-to-disable-systemd-resolved-and-resolve-dns-with-dnsmasq>

Basé sur « [How to avoid conflicts between dnsmasq and systemd-resolved?](#) » par [stackexchange.com](#).



¹⁾

C'est important, car resolv.conf est par défaut un lien symbolique vers /run/systemd/resolve/stub-resolv.conf. Si vous ne supprimez pas le lien symbolique, le fichier sera écrasé par systemd au redémarrage. NetworkManager (NM) vérifie également s'il s'agit d'un lien symbolique pour détecter la configuration de systemd-resolved.

²⁾

Par défaut, le fichier de configuration de dnsmasq est **/etc/dnsmasq.conf**, dont toutes les lignes sont commentées. En n'y touchant pas, on facilite les mises à jour.

³⁾

Name System Look Up

From:

<https://nfrappe.fr/doc/> - Documentation du Dr Nicolas Frappé

Permanent link:

<https://nfrappe.fr/doc/doku.php?id=logiciel:internet:dnsmasq:start>



Last update: 2022/11/08 19:27