

Logiciel

# Utilisation du plug-in DNSMasq de NetworkManager



Le plugin dnsmasq est un joyau caché de NetworkManager.

Lors de l'utilisation du plugin, au lieu d'utiliser n'importe quel serveur de noms DNS distribué par DHCP, NetworkManager configurera une copie locale de dnsmasq qui peut être personnalisée.

Pourquoi faire cela? Pour moi personnellement, j'ai deux cas d'utilisation:

Tout d'abord, sur mon ordinateur portable, j'exécute une installation complète d'OpenShift à des fins de test. Pour que cela fonctionne, j'ai vraiment besoin de pouvoir ajouter des enregistrements DNS. Je peux exécuter un dnsmasq local sans NetworkManager, mais cette configuration est plus facile que de gérer la mienne.

Deuxièmement, lorsque je suis à la maison, je souhaite toujours utiliser le DNS de mon réseau domestique lorsque je suis sur VPN. De nombreux VPN sont configurés pour acheminer uniquement le trafic spécifique via le tunnel VPN et laisser ma route par défaut en place. Cela signifie que je peux accéder à l'imprimante de mon réseau local et toujours me connecter aux ressources sur le VPN. C'est très bien, car cela signifie que je peux toujours accéder à mon imprimante réseau ou écouter de la musique depuis mon serveur multimédia tout en travaillant. Cependant, la connexion VPN écrase mon resolv.conf avec les serveurs DNS du réseau VPN. Par conséquent, le DNS de mon réseau domestique n'est plus accessible.

Le plugin dnsmasq résout ce problème en exécutant un serveur dnsmasq local contrôlé par NetworkManager.

Mon resolv.conf pointe toujours vers localhost.

Pour les enregistrements définis localement (par exemple pour mon cluster OpenShift), dnsmasq les résout correctement.

En utilisant une configuration dnsmasq plus avancée, je peux transférer de manière sélective des requêtes pour certains domaines vers des serveurs spécifiques (par exemple, pour toujours résoudre correctement les hôtes de mon réseau domestique).

Et pour toutes les autres demandes, dnsmasq transmettra aux serveurs DNS associés à mon réseau actuel ou VPN.

Voici comment le configurer dans Fedora 29:

Pour certains contextes, mon domaine sur mon ordinateur portable s'appelle «laplab» et mon domaine personnel est «.homelab».

À la maison, mon serveur DNS est 172.31.0.1.

Pour les entrées DNS dans laplab, la plupart de celles-ci sont définies dans /etc/hosts. dnsmasq peut alors les lire.

J'ai également quelques entrées DNS supplémentaires définies pour un DNS générique et quelques alias.

Voici les cinq fichiers qui doivent être ajoutés. Les fichiers dans dnsmasq.d pourraient être combinés, mais sont divisés pour mieux montrer l'exemple.

- /etc/NetworkManager/conf.d/00-use-dnsmasq.conf :

[/etc/NetworkManager/conf.d/00-use-dnsmasq.conf](#)

```
$ # /etc/NetworkManager/conf.d/00-use-dnsmasq.conf
#
# This enabled the dnsmasq plugin.
[main]
dns=dnsmasq
```

- /etc/NetworkManager/dnsmasq.d/00-homelab.conf :

[/etc/NetworkManager/dnsmasq.d/00-homelab.conf](#)

```
# /etc/NetworkManager/dnsmasq.d/00-homelab.conf
#
# This file directs dnsmasq to forward any request to resolve
# names under the .homelab domain to 172.31.0.1, my
# home DNS server.
server=/homelab/172.31.0.1
```

- /etc/NetworkManager/dnsmasq.d/01-laplab.conf :

[etc/NetworkManager/dnsmasq.d/01-laplab.conf](#)

```
# /etc/NetworkManager/dnsmasq.d/01-laplab.conf
# This file sets up the local lablab domain and
# defines some aliases and a wildcard.
local=/laplab/

# The below defines a Wildcard DNS Entry.
address=/.ose.laplab/192.168.101.125

# Below I define some host names. I also pull in
address=/openshift.laplab/192.168.101.120
address=/openshift-int.laplab/192.168.101.120
```

- /etc/NetworkManager/dnsmasq.d/02-add-hosts.conf :

[/etc/NetworkManager/dnsmasq.d/02-add-hosts.conf](#)

```
# /etc/NetworkManager/dnsmasq.d/02-add-hosts.conf
# By default, the plugin does not read from /etc/hosts.
# This forces the plugin to slurp in the file.
#
# If you didn't want to write to the /etc/hosts file. This
could
# be pointed to another file.
#
addn-hosts=/etc/hosts
```

- /etc/hosts :

[/etc/hosts](#)

```
# /etc/hosts
#
# The hostnames I define in that will be brought in and
resolvable
# because of the config in the 02-add-hosts.conf file.
#
127.0.0.1    localhost localhost.localdomain
::1         localhost localhost.localdomain

# Notice that my hosts be in the .laplab domain, like as
configured
# in 01-laplab.conf file
192.168.101.120 ose-lap-jumphost.laplab
192.168.101.128 ose-lap-node1.laplab

# Name not in .laplab will also get picked up. So be careful
# defining items here.
172.31.0.88    overwrite.public.domain.com
```

Une fois tous ces fichiers en place, redémarrez NetworkManager avec

```
$ systemctl restart NetworkManager
```

.

Si tout fonctionne correctement, vous devriez voir que votre resolv.conf pointe vers 127.0.0.1 et un nouveau processus dnsmasq est apparu.

```
$ ps -ef | grep dnsmasq
dnsmasq  1835  1188  0 08:01 ?          00:00:00 /usr/sbin/dnsmasq --no-
resolv
--keep-in-foreground --no-hosts --bind-interfaces --pid-
file=/var/run/NetworkManager/dnsmasq.pid
```

```
--listen-address=127.0.0.1 --cache-size=400 --clear-on-reload --conf-file=/dev/null  
--proxy-dnssec --enable-dbus=org.freedesktop.NetworkManager.dnsmasq  
--conf-dir=/etc/NetworkManager/dnsmasq.d
```

```
$ cat /etc/resolv.conf  
# Generated by NetworkManager  
nameserver 127.0.0.1
```

```
$ host ose-lap-jumphost.laplab  
ose-lap-jumphost.laplab has address 192.168.101.120
```

Cette configuration survivra aux redémarrages et, lors de mes tests, fonctionne avec presque tous les réseaux et VPN avec lesquels je l'ai essayé.

## Pré-requis

## Installation

## Configuration

## Utilisation

## Désinstallation

## Voir aussi

- (en) <https://fedoramagazine.org/using-the-networkmanagers-dnsmasq-plugin/>

---

Basé sur « *Using the NetworkManager's DNSMasq plugin* » par Clark Hale.

From: <http://doc.nfrappe.fr/> - **Documentation du Dr Nicolas Frappé**

Permanent link: <http://doc.nfrappe.fr/doku.php?id=logiciel:internet:dnsmaq:ubuntu18.04:nwm:start> 

Last update: **2022/08/13 22:27**

