Logiciel

Dnsmasq.conf : options

Format

Une option par ligne :

```
<option>=<valeur>
```

les options valides sont les mêmes que celles sur la ligne de commande.

Pour plus de détails, voir

```
...@...:~$ /usr/sbin/dnsmasq --help
```

ou

```
...@...:~$ man 8 dnsmasq
```

Options

port

> port=5353
>> Écouter sur ce port au lieu du port DNS standard (53).
>> Un réglage à zéro désactive complètement la fonction DNS, ne laissant que DHCP et/ou TFTP.

domain-needed et bogus-priv
> filtrer les requêtes auxquelles le DNS public ne peut pas répondre et qui chargent inutilement les serveurs.

> domain-needed
>> Pas de nom simple (sans point ni partie de domaine)
> bogus-priv
>> Pas d'adresse dans les espaces d'adressage non routés.

conf-file et dnssec
> Activer la validation DNSSEC et la mise en cache

> conf-file=%%PREFIX%%/share/dnsmasq/trust-anchors.conf
> dnssec

dnssec-check-unsigned
> vérifier qu'une réponse non signée est valide.

filterwin2k

Filtrer les requêtes DNS inutiles provenant de Windows qui peuvent être déclenchées.

resolv-file=

Modifiez cette ligne si vous voulez que DNS récupère ses serveurs d'un autre endroit que /etc/resolv.conf

strict-order

Essayer chaque requête avec chaque serveur dans l'ordre strict d'apparition dans /etc/resolv.conf

no-resolv

Pour que dnsmasq récupère ses serveurs à partir de ce fichier (voir ci-dessous) au lieu de /etc/resolv.conf ou tout autre fichier

no-poll

Ne pas interroger /etc/resolv.conf ou d'autres fichiers resolv pour chercher les changements et les relire

server

server=/localnet/192.168.0.1

Ajoutez ici d'autres serveurs de noms, avec des spécifications de domaine pour des domaines non publics.

server=/3.168.192.in-addr.arpa/10.1.2.3

Exemple de routage des requêtes PTR vers les serveurs de noms qui enverra toutes les requêtes adresse→nom pour 192.168.3/24 au serveur de noms 10.1.2.3

server=10.1.2.3@eth1

Vous pouvez contrôler comment dnsmasq parle à un serveur : ceci force les requêtes vers 10.1.2.3 à être acheminées via eth1

server=10.1.2.3@192.168.1.1#55

adresse source (c'est-à-dire locale) utilisée pour parler au port 55 de 10.1.2.3 à 192.168.1.1 (il doit y avoir une interface avec cette adresse IP sur la machine, évidemment).

local

local=/localnet/

Ajoutez ici des domaines locaux, les requêtes pour ces domaines ne sont traitées qu'à partir de /etc/hosts ou de DHCP

address

address=/double-click.net/127.0.0.1

Domaine dont vous voulez forcer l'adresse IP.
Cet exemple envoie n'importe quel hôte de double-click.net à un serveur Web local.

address=/www.thekelleys.org.uk/fe80::20d:60ff:fe36:f83

–address (et –server) fonctionnent également avec les adresses IPv6.

ipset

ipset=/yahoo.com/google.com/vpn,search

Ajoutez les adresses IP de toutes les requêtes à yahoo.com, google.com et leurs sous-domaines au VPN et aux ipsets de recherche

server
user et group

> Si vous voulez que dnsmasq change uid et gid en autre chose que la valeur par défaut,
> modifiez les lignes suivantes :

> user=
> group=

interface, except-interface, listen-address, no-dhcp-interface

> interface=
>> N'écouter les requêtes DHCP et DNS que sur les interfaces spécifiées (par
>> exemple eth0)
>> Répétez la ligne pour plusieurs interfaces.
> except-interface=
>> interface sur laquelle ne pas écouter
> listen-address=
>> interface sur laquelle écouter par adresse (n'oubliez pas d'inclure 127.0.0.1
>> si vous l'utilisez.)
> no-dhcp-interface=
>> If you want dnsmasq to provide only DNS service on an interface, configure
>> it as shown above, and then use the following line to disable DHCP and TFTP
>> on it.

bind-interfaces

> On systems which support it, dnsmasq binds the wildcard address, even when it is
> listening on only some interfaces. It then discards requests that it shouldn't reply to. This
> has the advantage of working even when interfaces come and go and change address. If
> you want dnsmasq to really bind only the interfaces it is listening on, uncomment this
> option. About the only time you may need this is when running another nameserver on
> the same machine.

no-hosts

> If you don't want dnsmasq to read /etc/hosts, uncomment the following line.

addn-hosts=/etc/banner_add_hosts

> if you want it to read another file, as well as /etc/hosts, use this.

expand-hosts

> Set this (and domain: see below) if you want to have a domain automatically added to
> simple names in a hosts-file.

domain

> domain=thekelleys.org.uk
>> Set the domain for dnsmasq. this is optional, but if it is set, it does the
>> following things.
>> 1) Allows DHCP hosts to have fully qualified domain names, as long as the
>> domain part matches this setting.
>> 2) Sets the "domain" DHCP option thereby potentially setting the domain of
>> all systems configured by DHCP
>> 3) Provides the domain part for "expand-hosts"
> domain=wireless.thekelleys.org.uk,192.168.2.0/24
>> Set a different domain for a particular subnet
> domain=reserved.thekelleys.org.uk,192.68.3.100,192.168.3.200

Same idea, but range rather then subnet

dhcp-range

dhcp-range=192.168.0.50,192.168.0.150,12h
> Uncomment this to enable the integrated DHCP server, you need to supply the range of addresses available for lease and optionally a lease time. If you have more than one network, you will need to repeat this for each network on which you want to supply DHCP service.

dhcp-range=192.168.0.50,192.168.0.150,255.255.255.0,12h
> This is an example of a DHCP range where the netmask is given. This is needed for networks we reach the dnsmasq DHCP server via a relay agent. If you don't know what a DHCP relay agent is, you probably don't need to worry about this.

dhcp-range=set:red,192.168.0.50,192.168.0.150
> This is an example of a DHCP range which sets a tag, so that some DHCP options may be set only for this network.

dhcp-range=tag:green,192.168.0.50,192.168.0.150,12h
> Use this DHCP range only when the tag "green" is set.

dhcp-range=192.168.0.0,static
> Specify a subnet which can't be used for dynamic address allocation, is available for hosts with matching –dhcp-host lines. Note that dhcp-host declarations will be ignored unless there is a dhcp-range of some type for the subnet in question. In this case the netmask is implied (it comes from the network configuration on the machine running dnsmasq) it is possible to give an explicit netmask instead.

dhcp-range=1234::2, 1234::500, 64, 12h
> Enable DHCPv6. Note that the prefix-length does not need to be specified and defaults to 64 if missing/

dhcp-range=1234::, ra-only
> Do Router Advertisements, BUT NOT DHCP for this subnet.

dhcp-range=1234::, ra-names
> Do Router Advertisements, BUT NOT DHCP for this subnet, also try and add names to the DNS for the IPv6 address of SLAAC-configured dual-stack hosts. Use the DHCPv4 lease to derive the name, network segment and MAC address and assume that the host will also have an IPv6 address calculated using the SLAAC algorithm.

dhcp-range=1234::, ra-only, 48h
> Do Router Advertisements, BUT NOT DHCP for this subnet.
> Set the lifetime to 46 hours. (Note: minimum lifetime is 2 hours.)

dhcp-range=1234::2, 1234::500, slaac
> Do DHCP and Router Advertisements for this subnet. Set the A bit in the RA so that clients can use SLAAC addresses as well as DHCP ones.

dhcp-range=1234::, ra-stateless
> Do Router Advertisements and stateless DHCP for this subnet. Clients will not get addresses from DHCP, but they will get other configuration information.
> They will use SLAAC for addresses.

dhcp-range=1234::, ra-stateless, ra-names
> Do stateless DHCP, SLAAC, and generate DNS names for SLAAC addresses from DHCPv4 leases.

enable-ra

Do router advertisements for all subnets where we're doing DHCPv6 Unless overridden by ra-stateless, ra-names, et al, the router advertisements will have the M and O bits set, so that the clients get addresses and configuration from DHCPv6, and the A bit reset, so the clients don't use SLAAC addresses.

dhcp-host

Supply parameters for specified hosts using DHCP. There are lots of valid alternatives, so we will give examples of each. Note that IP addresses DO NOT have to be in the range given above, they just need to be on the same network. The order of the parameters in these do not matter, it's permissible to give name, address and MAC in any order.

dhcp-host=11:22:33:44:55:66,192.168.0.60
> Always allocate the host with Ethernet address 11:22:33:44:55:66 The IP address 192.168.0.60

dhcp-host=11:22:33:44:55:66,fred
> Always set the name of the host with hardware address 11:22:33:44:55:66 to be "fred"

dhcp-host=11:22:33:44:55:66,fred,192.168.0.60,45m
> Always give the host with Ethernet address 11:22:33:44:55:66 the name fred and IP address 192.168.0.60 and lease time 45 minutes

dhcp-host=11:22:33:44:55:66,12:34:56:78:90:12,192.168.0.60
> Give a host with Ethernet address 11:22:33:44:55:66 or 12:34:56:78:90:12 the IP address 192.168.0.60. Dnsmasq will assume that these two Ethernet interfaces will never be in use at the same time, and give the IP address to the second, even if it is already in use by the first. Useful for laptops with wired and wireless addresses.

dhcp-host=bert,192.168.0.70,infinite
> Give the machine which says its name is "bert" IP address 192.168.0.70 and an infinite lease

dhcp-host=id:01:02:02:04,192.168.0.60
> Always give the host with client identifier 01:02:02:04 the IP address 192.168.0.60

dhcp-host=id:ff:00:00:00:00:00:02:00:00:02:c9:00:f4:52:14:03:00:28:05:81,192.168.0.61
> Always give the InfiniBand interface with hardware address 80:00:00:48:fe:80:00:00:00:00:00:00:f4:52:14:03:00:28:05:81 the ip address 192.168.0.61. The client id is derived from the prefix ff:00:00:00:00:00:02:00:00:02:c9:00 and the last 8 pairs of hex digits of the hardware address.

dhcp-host=id:marjorie,192.168.0.60
> Always give the host with client identifier "marjorie" the IP address 192.168.0.60

dhcp-host=judge
> Enable the address given for "judge" in /etc/hosts to be given to a machine presenting the name "judge" when it asks for a DHCP lease.

dhcp-host=11:22:33:44:55:66,ignore
> Never offer DHCP service to a machine whose Ethernet address is 11:22:33:44:55:66

dhcp-host=11:22:33:44:55:66,id:*
> Ignore any client-id presented by the machine with Ethernet address 11:22:33:44:55:66. This is useful to prevent a machine being treated differently when running under different OS's or between PXE boot and OS

boot.
dhcp-host=11:22:33:44:55:66,set:red
> Send extra options which are tagged as "red" to the machine with Ethernet
> address 11:22:33:44:55:66

dhcp-host=11:22:33:*:*:*,set:red
> Send extra options which are tagged as "red" to any machine with Ethernet
> address starting 11:22:33:

dhcp-host=id:00:01:00:01:16:d2:83:fc:92:d4:19:e2:d8:b2, fred, [1234::5]
> Give a fixed IPv6 address and name to client with DUID
> 00:01:00:01:16:d2:83:fc:92:d4:19:e2:d8:b2
> Note the MAC addresses CANNOT be used to identify DHCPv6 clients.
> Note also that the [] around the IPv6 address are obligatory.

dhcp-ignore=tag:!known
> Ignore any clients which are not specified in dhcp-host lines or /etc/ethers. Equivalent to
> ISC "deny unknown-clients".
> This relies on the special "known" tag which is set when a host is matched.

dhcp-vendorclass=set:red,Linux
> Send extra options which are tagged as "red" to any machine whose DHCP vendorclass
> string includes the substring "Linux"

dhcp-userclass=set:red,accounts
> Send extra options which are tagged as "red" to any machine one of whose DHCP
> userclass strings includes the substring "accounts"

dhcp-mac=set:red,00:60:8C:*:*:*
> Send extra options which are tagged as "red" to any machine whose MAC address
> matches the pattern.

read-ethers
> If this line is uncommented, dnsmasq will read /etc/ethers and act on the ethernet-
> address/IP pairs found there just as if they had been given as –dhcp-host options. Useful
> if you keep MAC-address/host mappings there for other purposes.

dhcp-option
> Send options to hosts which ask for a DHCP lease.
> See RFC 2132 for details of available options.
> Common options can be given to dnsmasq by name:

> > run "dnsmasq –help dhcp" to get a list.

> Note that all the common settings, such as netmask and broadcast address, DNS server
> and default route, are given sane defaults by dnsmasq. You very likely will not need any
> dhcp-options. If you use Windows clients and Samba, there are some options which are
> recommended, they are detailed at the end of this section.

> dhcp-option=3,1.2.3.4
> > Override the default route supplied by dnsmasq, which assumes the router is
> > the same machine as the one running dnsmasq.

> dhcp-option=option:router,1.2.3.4
> > Do the same thing, but using the option name

> dhcp-option=3
> > Override the default route supplied by dnsmasq and send no default route at
> > all. Note that this only works for the options sent by default (1, 3, 6, 12, 28)
> > the same line will send a zero-length option for all other option numbers.

dhcp-option=option:ntp-server,192.168.0.4,10.10.0.5

> Set the NTP time server addresses to 192.168.0.4 and 10.10.0.5

dhcp-option=option6:dns-server,[1234::77],[1234::88]

> Send DHCPv6 option. Note [] around IPv6 addresses.

dhcp-option=option6:dns-server,[::],[1234::88]

> Send DHCPv6 option for namservers as the machine running dnsmasq and another.

dhcp-option=option6:information-refresh-time,6h

> Ask client to poll for option changes every six hours. (RFC4242)

dhcp-option=option:T1,1m

> Set option 58 client renewal time (T1). Defaults to half of the lease time if not specified. (RFC2132)

dhcp-option=option:T2,2m

> Set option 59 rebinding time (T2). Defaults to 7/8 of the lease time if not specified. (RFC2132)

dhcp-option=42,0.0.0.0

> Set the NTP time server address to be the same machine as is running dnsmasq

dhcp-option=40,welly

> Set the NIS domain name to "welly"

dhcp-option=23,50

> Set the default time-to-live to 50

dhcp-option=27,1

> Set the "all subnets are local" flag

dhcp-option=128,e4:45:74:68:00:00
dhcp-option=129,NIC=eepro100

> Send the etherboot magic flag and then etherboot options (a string).

dhcp-option = tag:red, option:ntp-server, 192.168.1.1

> Specify an option which will only be sent to the "red" network (see dhcp-range for the declaration of the "red" network)
>
> Note that the tag: part must precede the option: part.

The following DHCP options

> set up dnsmasq in the same way as is specified for the ISC dhcpcd in http://www.samba.org/samba/ftp/docs/textdocs/DHCP-Server-Configuration.txt adapted for a typical dnsmasq installation where the host running dnsmasq is also the host running samba. you may want to uncomment some or all of them if you use Windows clients and Samba.
>
> dhcp-option=19,0
>
> > option ip-forwarding off
>
> dhcp-option=44,0.0.0.0
>
> > set netbios-over-TCP/IP nameserver(s) aka WINS server(s)
>
> dhcp-option=45,0.0.0.0
>
> > netbios datagram distribution server
>
> dhcp-option=46,8
>
> > netbios node type

dhcp-option=252,"\n"

> Send an empty WPAD option. This may be REQUIRED to get windows 7 to behave.

dhcp-option=option:domain-search,eng.apple.com,marketing.apple.com

Send RFC-3397 DNS domain search DHCP option. WARNING: Your DHCP client probably doesn't support this......

dhcp-option=121,192.168.1.0/24,1.2.3.4,10.0.0.0/8,5.6.7.8

Send RFC-3442 classless static routes (note the netmask encoding)

dhcp-option=vendor:PXEClient,1,0.0.0.0

Send vendor-class specific options encapsulated in DHCP option 43. The meaning of the options is defined by the vendor-class so options are sent only when the client supplied vendor class matches the class given here. (A substring match is OK, so "MSFT" matches "MSFT" and "MSFT 5.0"). This example sets the mtftp address to 0.0.0.0 for PXEClients.

dhcp-option=vendor:MSFT,2,1i

Send microsoft-specific option to tell windows to release the DHCP lease when it shuts down. Note the "i" flag, to tell dnsmasq to send the value as a four-byte integer - that's what microsoft wants. See http://technet2.microsoft.com/WindowsServer/en/library/a70f1bb7-d2d4-49f0-96d6-4b7414ecfaae1033.mspx?mfr=true

dhcp-option=vendor:Etherboot,60,"Etherboot"

Send the Encapsulated-vendor-class ID needed by some configurations of Etherboot to allow is to recognise the DHCP server.

dhcp-option-force

dhcp-option-force=208,f1:00:74:7e

Send options to PXELinux. Note that we need to send the options even though they don't appear in the parameter request list, so we need to use dhcp-option-force here. See http://syslinux.zytor.com/pxe.php#special for details. Magic number - needed before anything else is recognised

dhcp-option-force=209,configs/common

Configuration file name

dhcp-option-force=210,/tftpboot/pxelinux/files/

Path prefix

dhcp-option-force=211,30i

Reboot time. (Note 'i' to send 32-bit value)

dhcp-boot

dhcp-boot=pxelinux.0

Set the boot filename for netboot/PXE. You will only need this if you want to boot machines over the network and you will need a TFTP server; either dnsmasq's built-in TFTP server or an external one. (See below for how to enable the TFTP server.)

dhcp-boot=pxelinux,server.name,192.168.1.100

The same as above, but use custom tftp-server instead machine running dnsmasq

Boot for iPXE

The idea is to send two different filenames, the first loads iPXE, and the second tells iPXE what to load. The dhcp-match sets the ipxe tag for requests from iPXE.

dhcp-boot=undionly.kpxe

dhcp-match=set:ipxe,175 # iPXE sends a 175 option.
dhcp-boot=tag:ipxe,http://boot.ipxe.org/demo/boot.php

Encapsulated options for iPXE.
All the options are encapsulated within option 175

dhcp-option=encap:175, 1, 5b
priority code
dhcp-option=encap:175, 176, 1b
no-proxydhcp
dhcp-option=encap:175, 177, string
bus-id
dhcp-option=encap:175, 189, 1b
BIOS drive code
dhcp-option=encap:175, 190, user
iSCSI username
dhcp-option=encap:175, 191, pass
iSCSI password

dhcp-match

Test for the architecture of a netboot client.
PXE clients are supposed to send their architecture as option 93. (See RFC 4578)

dhcp-match=peecees, option:client-arch, 0 #x86-32
dhcp-match=itanics, option:client-arch, 2 #IA64
dhcp-match=hammers, option:client-arch, 6 #x86-64
dhcp-match=mactels, option:client-arch, 7 #EFI x86-64

pxe-prompt

pxe-prompt="What system shall I netboot?"
Do real PXE, rather than just booting a single file, this is an alternative to
dhcp-boot.
pxe-prompt="Press F8 for menu.", 60
or with timeout before first available action is taken:

pxe-service

pxe-service=x86PC, "Boot from local disk"
Available boot services. for PXE.
pxe-service=x86PC, "Install Linux", pxelinux
Loads <tftp-root>/pxelinux.0 from dnsmasq TFTP server.
pxe-service=x86PC, "Install Linux", pxelinux, 1.2.3.4
Loads <tftp-root>/pxelinux.0 from TFTP server at 1.2.3.4.
Beware this fails on old PXE ROMS.
pxe-service=x86PC, "Install windows from RIS server", 1
Use bootserver on network, found my multicast or broadcast.
pxe-service=x86PC, "Install windows from RIS server", 1, 1.2.3.4
Use bootserver at a known IP address.

multicast-FTP
If you have multicast-FTP available, information for that can be passed in a similar way

using options 1 to 5. See page 19 of
http://download.intel.com/design/archives/wfm/downloads/pxespec.pdf

enable-tftp
> Enable dnsmasq's built-in TFTP server

tftp-root=/var/ftpd
> Set the root directory for files available via FTP.

tftp-no-fail
> Do not abort if the tftp-root is unavailable

tftp-secure
> Make the TFTP server more secure: with this set, only files owned by the user dnsmasq is running as will be send over the net.

tftp-no-blocksize
> This option stops dnsmasq from negotiating a larger blocksize for TFTP transfers. It will slow things down, but may rescue some broken TFTP clients.

dhcp-boot

dhcp-boot=tag:red,pxelinux.red-net
> Set the boot file name only when the "red" tag is set.

dhcp-boot=/var/ftpd/pxelinux.0,boothost,192.168.0.3
> An example of dhcp-boot with an external TFTP server: the name and IP address of the server are given after the filename.
> Can fail with old PXE ROMS. Overridden by –pxe-service.

dhcp-boot=/var/ftpd/pxelinux.0,boothost,tftp_server_name
> If there are multiple external tftp servers having a same name (using /etc/hosts) then that name can be specified as the tftp_servername (the third option to dhcp-boot) and in that case dnsmasq resolves this name and returns the resultant IP addresses in round robin fashion. This facility can be used to load balance the tftp load among a set of servers.

dhcp-lease

dhcp-lease-max=150
> Set the limit on DHCP leases, the default is 150

dhcp-leasefile=/var/lib/misc/dnsmasq.leases
> The DHCP server needs somewhere on disk to keep its lease database.
> This defaults to a sane location, but if you want to change it, use the line below.

dhcp-authoritative
> Set the DHCP server to authoritative mode. In this mode it will barge in and take over the lease for any client which broadcasts on the network, whether it has a record of the lease or not. This avoids long timeouts when a machine wakes up on a new network. DO NOT enable this if there's the slightest chance that you might end up accidentally configuring a DHCP server for your campus/company accidentally. The ISC server uses the same option, and this URL provides more information: http://www.isc.org/files/auth.html

dhcp-rapid-commit
> Set the DHCP server to enable DHCPv4 Rapid Commit Option per RFC 4039.
> In this mode it will respond to a DHCPDISCOVER message including a Rapid Commit option with a DHCPACK including a Rapid Commit option and fully committed address and configuration information. This must only be enabled if either the server is the only

server for the subnet, or multiple servers are present and they each commit a binding for all clients.

dhcp-script=/bin/echo

Run an executable when a DHCP lease is created or destroyed.
The arguments sent to the script are "add" or "del", then the MAC address, the IP address and finally the hostname if there is one.

cache-size=150

Set the cachesize here.

no-negcache

If you want to disable negative caching, uncomment this.

local-ttl= Normally responses which come from /etc/hosts and the DHCP lease file have Time-To-Live set as zero, which conventionally means do not cache further. If you are happy to trade lower load on the server for potentially stale date, you can set a time-to-live (in seconds) here.

bogus-nxdomain=64.94.110.11

If you want dnsmasq to detect attempts by Verisign to send queries to unregistered .com and .net hosts to its sitefinder service and have dnsmasq instead return the correct NXDOMAIN response, uncomment this line. You can add similar lines to do the same for other registries which have implemented wildcard A records.

alias

If you want to fix up DNS results from upstream servers, use the alias option. This only works for IPv4.

alias=1.2.3.4,5.6.7.8

This alias makes a result of 1.2.3.4 appear as 5.6.7.8
alias=1.2.3.0,5.6.7.0,255.255.255.0

and this maps 1.2.3.x to 5.6.7.x
alias=192.168.0.10-192.168.0.40,10.0.0.0,255.255.255.0

and this maps 192.168.0.10→192.168.0.40 to 10.0.0.10→10.0.0.40

MX records

Change these lines if you want dnsmasq to serve MX records.

mx-host=maildomain.com,servermachine.com,50

Return an MX record named "maildomain.com" with target servermachine.com and preference 50
mx-target=servermachine.com

Set the default target for MX records created using the localmx option.
localmx

Return an MX record pointing to the mx-target for all local machines.
selfmx

Return an MX record pointing to itself for all local machines.

srv-host

Change the following lines if you want dnsmasq to serve SRV records. These are useful if you want to serve ldap requests for Active Directory and other windows-originated DNS requests.
See RFC 2782.
You may add multiple srv-host lines.
The fields are <name>,<target>,<port>,<priority>,<weight>
If the domain part if missing from the name (so that is just has the service and protocol sections) then the domain given by the domain= config option is used. (Note that expand-hosts does not need to be set for this to work.)

srv-host=_ldap._tcp.example.com,ldapserver.example.com,389
> A SRV record sending LDAP for the example.com domain to ldapserver.example.com port 389

domain=example.com

srv-host=_ldap._tcp,ldapserver.example.com,389
> A SRV record sending LDAP for the example.com domain to ldapserver.example.com port 389 (using domain=)

srv-host=_ldap._tcp.example.com,ldapserver.example.com,389,1
srv-host=_ldap._tcp.example.com,ldapserver.example.com,389,2
> Two SRV records for LDAP, each with different priorities

srv-host=_ldap._tcp.example.com
> A SRV record indicating that there is no LDAP server for the domain example.com

ptr-record=_http._tcp.dns-sd-services,"New Employee Page._http._tcp.dns-sd-services"
> The following line shows how to make dnsmasq serve an arbitrary PTR record. This is useful for DNS-SD. (Note that the domain-name expansion done for SRV records _does_not occur for PTR records.)

txt-record
> Change the following lines to enable dnsmasq to serve TXT records.
> These are used for things like SPF and zeroconf. (Note that the domain-name expansion done for SRV records _does_not occur for TXT records.)

> txt-record=example.com,"v=spf1 a -all"
> > Example SPF.
> txt-record=_http._tcp.example.com,name=value,paper=A4
> > Example zeroconf

cname=bertand,bert
> Provide an alias for a "local" DNS name. Note that this _only_ works for targets which are names from DHCP or /etc/hosts. Give host "bert" another name, bertrand

log-queries
> For debugging purposes, log each DNS query as it passes through dnsmasq.

log-dhcp
> Log lots of extra information about DHCP transactions.

conf-file=/etc/dnsmasq.more.conf

conf-dir=/etc/dnsmasq.d
> Include another lot of configuration options.

conf-dir=/etc/dnsmasq.d,.bak
> Include all the files in a directory except those ending in .bak

conf-dir=/etc/dnsmasq.d/,*.conf
> Include all files in a directory which end in .conf

dhcp-name-match=set:wpad-ignore,wpad
dhcp-ignore-names=tag:wpad-ignore
> If a DHCP client claims that its name is "wpad", ignore that.
> This fixes a security hole. see CERT Vulnerability VU#598349

Voir aussi

- **(fr)** http://Article

Basé sur « *Article* » par Auteur.

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

Permanent link:
http://doc.nfrappe.fr/doku.php?id=logiciel:internet:dnsmasq:config:start3

Last update: **2022/11/08 19:27**