dnsmasq Reference Guide

Grow with Google: IT Support Certificate

Sections

Introduction to dnsmasq

Dnsmasq is a lightweight program that operates on Linux-based operating systems. It provides Domain Name System (DNS) services like caching as well as functioning as a Dynamic Host Configuration Protocol (DHCP) server. DNS services and the DHCP server can be deployed together or separately depending on your requirements.

dnsmasq Configuration

1. **Install**

   You can install dnsmasq from the apt repository. Open a shell and run:

   ```
   sudo apt update
   sudo apt install dnsmasq
   ```

   After this step you have a simple functioning DNS server.

2. **Edit configuration file**

   By default, DNS is configured to forward all requests to your system's default DNS settings. To make any changes to the default settings you'll need to open the configuration file at /etc/dnsmasq.conf with a text editor.

Inside the configuration file you will find well commented and explained options. You can simply remove the comments to enable many of the default configuration options.

3. **Assign a network to dnsmasq**

Dnsmasq needs to be told where to look for DHCP and DNS requests. You can do so by setting the `listen-address` option to its LAN IP addresses (remember to include **127.0.0.1**) or you can restrict the interface dnsmasq listens on using the interface option (add more lines for more than one interface).

For example:
```
listen-address=::1,127.0.0.1,192.168.56.10
or,
interface=eth0
```

4. **Set a domain**

If you want to have a domain automatically added to simple names in a hosts-file, uncomment the expand-hosts option. To set a specific domain for dnsmasq, which means DHCP clients will have fully qualified domain names as long as the set domain is matched, and sets the "**domain**" DHCP option for all clients.

For example:
```
expand-hosts
domain=example.lan
```

5. **Test the syntax**

Save the file and test the syntax for the configuration file with the following command:
```
dnsmasq --test
```

Configure DHCP

1. **Turn on the DHCP**

By default, DHCP is turned off. This is a good thing, as you could bring down whatever network you are connected to if you are not careful. To enable it you must edit the /etc/dnsmasq.conf file and remove the comment on dhcp-range. Removing this comment will assign addresses to clients in the defined range. You can also define a lease time for each address. The default lease time is 12h.

For example:
```
dhcp-range=192.168.0.50,192.168.0.150,12h
```
If you have two networks you want DHCP to manage you can simply add them sequentially.

For example:
```
dhcp-range=eth0,192.168.100.100,192.168.100.199,12h
dhcp-range=eth1,192.168.200.100,192.168.200.199,12h
```

2. **Test new clients**

Save the file and use the command:
```
dnsmasq restart
```

Check to see if new clients are auto-configure themselves.

dnsmasq Commands

When using the following commands, you can use either the short or long version. Both versions are listed here separated by a comma.

--E,--expand-hosts
Add the domain to simple names in /etc/hosts in the same way as DHCP-derived names

Example: You want to prepare your network to use your own domain example.com.

-d, --no-daemon
Use debug mode: don't fork to the background, don't write a pid file, don't change user id, generate a complete cache dump on receipt on SIGUSR1, log to stderr as well as syslog, don't fork new processes to handle TCP queries

> Example: You need to use debug mode.

-q, --log-queries
Log the results of DNS queries handled by dnsmasq

> Example: You want to log queries in debug mode.

-v, --version
Prints the version number

> Example: You need to know what version of dnsmasq you are using.

-p ,--port=<port>
Listen on <port> instead of the standard DNS port (53). Setting this to zero completely disables the DNS function, leaving only DHCP and/or TFTP.

> **<port>**
> Specifies which port you want to use
>
> Example: You want to disable DNS so you use the argument: `-port=0`

-i, --interface=<interface name>
Listen only on the specified interface(s). Dnsmasq automatically adds the loopback (local) interface to the list of interfaces to use when the --interface option is used.

> **<interface name>**
> Specifies what network interface you want to use (e.g. eth0, wlan).
>
> Example: If you only want dnsmasq to listen on eth0 then you can use:
> `--interface=eth0`

-2, --no-dhcp-interface=<interface name>

Do not provide DHCP or TFTP on the specified interface, but do provide DNS service

> **`<interface name>`**
> Specifies what network interface you want to use (e.g. eth0, wlan).
>
> Example: You want to disable DHCP on your WLAN you can use:
> `--no-dhcp-interface=wlan0`

`-a, --listen-address=<ipaddr>`
Listen on the given IP address.

> **`<ipaddr>`**
> Specifies the IP address you want dnsmasq to listen on.
>
> Example: You want dnsmasq to listen to your IP address and the loopback so you use: `-listen-address=127.0.0.1,192.168.56.10`
>
> Both `--interface` and `--listen-address` options may be given. If both are given, the set of both interfaces and addresses is used. If no `--interface` option is given, but `--listen-address` is, dnsmasq will not automatically listen on the loopback interface. To achieve this, its IP address, 127.0.0.1, must be explicitly given as a `--listen-address` option.

`--clear-on-reload`
Whenever `/etc/resolv.conf` is re-read, clear the DNS cache. This is useful when new nameservers may have different data than that held in cache.

> Example: You have new nameservers and want to prepare to restart the network.

`-F,--dhcp-range=[network-id,<start-addr>,<end-addr>,<default lease time>]`
Enable the DHCP server. Addresses will be given out from the range `<start-addr>` to `<end-addr>` and from statically defined addresses given in dhcp-host options. If the lease time is given, then leases will be given for that length of time.

> **`network-id`**

Specifies which interface you want to hand out addresses to (e.g. eth0, wlan).

`<start-addr>`
Specifies the beginning of the range of addresses you want to hand out.

`<end-addr>`
Specifies the end of the range of addresses you want to hand out.

`<default lease time>`
Specifies the length of time assigned IP addresses will be valid for before needed to renew the lease. The lease time is in seconds, or minutes (eg 45m) or hours (eg 1h).

Example: You need to enable DHCP on your network so you use:
```
dhcp-range=eth0, 192.168.0.50,192.168.0.150,12h
```