# Packet DNS_DHCP - Hostnames, DNS- and DHCP-Server
# Version 3.10.18

Peter Schiefer
email: team@fli4l.de

the fli4l-team
email: team@fli4l.de

September 15, 2019

# Contents

# 1 Documentation of packet DNS_DHCP

## 1.1 DNS_DHCP - Hostnames, DNS- and DHCP-Server as well as DHCP-Relay

### 1.1.1 Hostnames

#### Hosts

**OPT_HOSTS** The configuration of hostnames can be disabled by means of the optional variable `OPT_HOSTS`!

**HOST_N HOST_x_{attribute}** All hosts in the LAN should be described - with IP-address, name, aliasname and perhaps Mac-address for the dhcp-configuration. At first we have to set the number of computers with the variable `HOST_N`.

**Note:** Since version 3.4.0, the entry for the router comes from the information in the `<config>/base.txt`. For additional aliasnames, see `HOSTNAME_ALIAS_N` (Page **??**).

Then the attributes define the properties of the hosts. Here are some of the attributes required, e.g. IP address and name, the other options are optional.

**NAME** – Name of the n'th host

**IP4** – IP address (ipv4) of the n'th host

**IP6** – IP address (ipv6) of the n'th host (optional). If you use "auto", then the address will be computed automatically from an IPv6 prefix (with /64 subnet mask) and the MAC address of the corresponding host, provided you activate OPT_IPV6. In order to make this work, you will have to set `HOST_x_MAC` (see below) and to properly configure the "ipv6" package.

**DOMAIN** – DNS domain of the n'th host (optional)

**ALIAS_N** – Number of aliasnames of the n'th host

**ALIAS_m** – m'th aliasname of the n'th host

**MAC** – MAC address of the n'th host

**DHCPTYP** – Assigning the IP address by DHCP depending on MAC or NAME (optional)

In the sample configuration file 4 hosts are configured, "client1", "client2", "client3" and "client4".

```
HOST_1_NAME='client1'                # 1st host: ip and name
HOST_1_IP4='192.168.6.1'
```

Alias names must be specified with complete domain.

The MAC address is optional and is only relevant if fli4l is used as a DHCP server additionally. This is explained below in the description of the optional package "OPT_-DHCP" Without use as a DHCP Server only the IP address, the name of the host and possibly the alias name are used. The MAC address is a 48-bit address and consists of 6 hex values seperated by a colon, for example

```
HOST_2_MAC='de:ad:af:fe:07:19'
```

*Note:* If fli4l is supplemented with the IPv6 packet, IPv6 addresses are not needed, if the MAC addresses of the hosts are present, because the IPv6 packet calculates the IPv6 addresses automatically (modified EUI-64). Of course, you can disable the automatic and use dedicated IPv6 addresses, if you wish.

#### Extra Hosts

**HOST_EXTRA_N HOST_EXTRA_x_NAME HOST_EXTRA_x_IP4 HOST_EXTRA_x_IP6**
Using these variables, you may add other hosts which are not members of the local domain e.g. hosts on the other side of a VPN.

### 1.1.2 DNS-Server

**OPT_DNS** To activate the DNS-server the variable OPT_DNS must be set to 'yes'.

If in the LAN no Windows machines are used or it has already a running DNS server, OPT_DNS can be set to 'no' and you may skip the rest in this section.

In doubt, set (Default setting): OPT_DNS='yes'

#### General DNS-options

**DNS_LISTEN_N DNS_LISTEN_x** If you chose OPT_DNS='yes', use DNS_LISTEN_N to set the number and DNS_LISTEN_1 up to DNS_LISTEN_N to specify the local IPs where **dnsmasq** accepts DNS-queries. If you set DNS_LISTEN_N to 0, **dnsmasq** answers DNS-queries on all local IPs. Only IPs of existing interfaces (ethernet, wlan ...) are allowed. Alternatively you can use ALIAS-Names here, i.e. IP_NET_1_IPADDR.

For all addresses specfied here ACCEPT rules will be created in the firewall's INPUT chains if PF_INPUT_ACCEPT_DEF='yes' and/or PF6_INPUT_ACCEPT_DEF='yes'. In case of DNS_LISTEN='0' rules allowing DNS access on *all* interfaces configured will be created.

**Important:** *If you want the DNS server to listen for interfaces dynamically added at runtime, such as VPN tunnel network interfaces, you should leave this array empty, otherwise the DNS server will not respond to DNS requests made through the VPN server.*

If in doubt, the default settings should be used.

**DNS_BIND_INTERFACES** If you only want to bind the DNS server to specific addresses via DNS_LISTEN_x *and* additionally want to bind *another* DNS server to *another* address, this option can be used to instruct the DNS server to *only* bind to the listed addresses.

By default, the DNS server binds to *all* interfaces and discards queries originating from addresses not configured. This has the advantage that the DNS server can also deal with interfaces added dynamically at runtime, but has the disadvantage that no alternative DNS server can run on the standard DNS port 53 at the same time. A use case for a second DNS server is if you want to run a slave DNS server like "yadifa" directly on the fli4l router. If you do not want to use the dnsmasq exclusively on the fli4l, you have to select the setting 'yes' and configure the IP addresses to be used for the dnsmasq via `DNS_LISTEN`.

**DNS_VERBOSE** Logging of DNS-queries: 'yes' or 'no'

For detailed messages from the DNS `DNS_VERBOSE` has to be set to yes. DNS-queries are logged to the syslog then. To see the messages you must set `OPT_SYSLOGD='yes'` (Page **??**) - see below.

**DNS_MX_SERVER** This variable indicates the hostname for the MX-record (Mail-Exchanger) for the domain defined in `DOMAIN_NAME`. A MTA (Mail"=Transport"=Agent, i.e. sendmail) on an internal server asks the DNS for a Mail-Exchanger for the destination domain of the mail beeing delivered.

**This is no mail-client autoconfiguration for i.e. Outlook! So please do not insert gmx.de here and wonder why Outlook does not work.**

**DNS_FORBIDDEN_N DNS_FORBIDDEN_x** Here you can provide domains, which are always beeing answered as "not existend".

Example:

```
DNS_FORBIDDEN_N='1'
DNS_FORBIDDEN_1='foo.bar'
```

In this case, a query for www.foo.bar is answered by an error. You can inhibit entire Top-Level-Domains in this way:

```
DNS_FORBIDDEN_1='de'
```

Then the name resolution for all hosts of the .de Topleveldomain is switched off.

**DNS_REDIRECT_N DNS_REDIRECT_x DNS_REDIRECT_x_IP** Here you can specify domains, which are beeing redirected to a specific IP.

Example:

```
DNS_REDIRECT_N='1'
DNS_REDIRECT_1='yourdom.dyndns.org'
DNS_REDIRECT_1_IP='192.168.6.200'
```

This redirects a query of yourdom.dyndns.org to IP 192.168.6.200.

**DNS_BOGUS_PRIV** If you set this variable to ‘yes‘, reverse-lookups for IP-Addresses of RFC1918 (Private Address Ranges) are not redirected to other DNS-servers but rather answered by the dnsmasq.

**DNS_FORWARD_PRIV_N DNS_FORWARD_PRIV_x** Sometimes you want to delegate the address resolution of some private subnets to the configured DNS server despite of an activated `DNS_BOGUS_PRIV`. This is necessary for example, if an uplink router manages private subnets. This array variable can be used for specifying the private subnets where address resolution should be delegated.

**DNS_FILTERWIN2K** If this is set to ’yes’ DNS queries of type SOA, SRV, and ANY will be blocked. Services using these queries will not work anymore without further configuration.
For example:

- XMPP (Jabber)
- SIP
- LDAP
- Kerberos
- Teamspeak3 (as of client-version 3.0.8)
- Minecraft (as of full version 1.3.1)
- domain controller discovery (Win2k)

For further information:

- Explanantion of DNS query types in general:
  http://en.wikipedia.org/wiki/List_of_DNS_record_types
- dnsmasq manpage:
  http://www.thekelleys.org.uk/dnsmasq/docs/dnsmasq-man.html
- SRV queries in detail:
  http://en.wikipedia.org/wiki/SRV_resource_record

**By setting this to ’no’, additionally forwarded DNS queries may cause unwanted dial-up connections or prevent existing ones from being closed. Particularly if you are using ISDN or UMTS connections additional costs may arise. You have to choose for yourself what’s more important to you.**

**DNS_FORWARD_LOCAL** By setting this variable to ’yes’ fli4l-routers may be configured to be in a domain by the name of DOMAIN_NAME=’example.local’ whose name resolution will be done by another name server specified by DNS_SPECIAL_x_DOMAIN=’example.local’.

**DNS_LOCAL_HOST_CACHE_TTL** defines the TTL (Time to live, in seconds) for entries defined in /etc/hosts as well as for hosts listed in DHCP. The default value for the fli4l-router is 60 seconds. Dnsmasq uses 0 as default and thus disables caching of DNS entries. The idea behind that is to reuse DHCP leases that are running out fastly and pass them on swiftly. However, if for example a local IMAP proxy queries the DNS entries several times per second this is a significant burden on the network. A compromise is a relatively short TTL of 60 seconds. Even without the short TTL 60 seconds a host can always

simply be switched off, so that the polling software has to deal with hosts not responding anyway.

**DNS_SUPPORT_IPV6** (optional)

Setting this optional variable to 'yes' enables the support for IPV6 Addresses of the DNS server.

### DNS Zone Configuration

Dnsmasq can also manage a DNS domain autonomously, being "authoritative" for it. Two things have to be done to achieve this: At first you have to specify which external (!) DNS name service points to your fli4l and on which network interface the resolution takes place. The specification of an external reference is required because the domain that is managed by fli4l will always be a subdomain of another domain. [1] The specification of the "outward" interface is important because the dnsmasq will behave different from other "inward" interfaces there: "Outwards" dnsmasq will never answer queries for names outside of its configured own domain. "Inwards" dnsmasq also acts as a DNS relay to the Internet to accomplish resolution of non-local names.

As the second thing you have to configure which networks can be reached from outside via name resolution. Of course only nets with public IP addresses can be specified because hosts with private addresses cannot be reached from outside.

Below the configuration will be described with an example. This example assumes IPv6 packets as well as a publicly routed IPv6 prefix; the latter can be provided i.e. by a 6in4 tunnel provider such as Hurricane Electric.

**DNS_AUTHORITATIVE** Specifying `DNS_AUTHORITATIVE='yes'` activates dnsmasq's authoritative mode. However, this is not enough, some more information must be given (see below).

Default Setting: `DNS_AUTHORITATIVE='no'`

Example: `DNS_AUTHORITATIVE='yes'`

**DNS_AUTHORITATIVE_NS** With this variable, the DNS name is configured by which the fli4l is referenced from outside using a DNS-NS record. This can also be a DNS name from a dynamic DNS service.

Example: `DNS_AUTHORITATIVE_NS='fli4l.noip.me'`

**DNS_AUTHORITATIVE_IPADDR** This variable configures the address resp. interface on which dnsmasq will answer DNS queries for your own domain authoritatively. Symbolic names like `IP_NET_2_IPADDR` are allowed. The dnsmasq can only answer authoritative on *one* address resp. interface.

Currently only permanently assigned addresses can be specified. Addresses derived only by a dial-in (for example, using a PPP connection), can not be used. This will be corrected in a later version of fli4l.

**Important:** *It should be noted that this should never be an address / interface connected to your own LAN, otherwise non-local names could not be resolved anymore!*

[1] We assume that noone uses a fli4l as a DNS root server...

Example: `DNS_AUTHORITATIVE_IPADDR='IP_NET_2_IPADDR'`

**DNS_ZONE_NETWORK_N DNS_ZONE_NETWORK_x** Specify the network addresses here for which the dnsmasq should resolve names authoritatively. Both forward (name to address) and reverse lookup (address to name) will work.

A complete example:

```
DNS_AUTHORITATIVE='yes'
DNS_AUTHORITATIVE_NS='fli4l.noip.me'
DNS_AUTHORITATIVE_IPADDR='IP_NET_2_IPADDR' # Uplink connected to eth1
DNS_ZONE_NETWORK_N='1'
DNS_ZONE_NETWORK_1='2001:db8:11:22::/64'   # local IPv6-LAN
```

It is assumed here that "2001:db8:11::/48" is a IPv6 prefix publicly routed to fli4l and that subnet 22 was chosen for the LAN.

### DNS Zone Delegation

**DNS_ZONE_DELEGATION_N DNS_ZONE_DELEGATION_x** There are special situations where the reference to one or more DNS server is useful, for example when using fli4l in an intranet without an Internet connection or a mix of these (intranet with an own DNS-server Internet connection in addition)

Imagine the following scenario:

- Circuit 1: Dial into the Internet
- Circuit 2: Dial into the Company network 192.168.1.0 (firma.de)

Then you set `ISDN_CIRC_1_ROUTE` to '0.0.0.0' and `ISDN_CIRC_2_ROUTE` to '192.168.1.0'. When accessing hosts with IP-Addresses with 192.168.1.x, fli4l will use circuit 2, otherwise circuit 1. But if the company network isn't public, it presumably has its own DNS server. Suppose the address of this DNS server would be 192.168.1.12 and the domain name would be " firma.de".

In this case you will write:

```
DNS_ZONE_DELEGATION_N='1'
DNS_ZONE_DELEGATION_1_UPSTREAM_SERVER_N='1'
DNS_ZONE_DELEGATION_1_UPSTREAM_SERVER_1_IP='192.168.1.12'
DNS_ZONE_DELEGATION_1_DOMAIN_N='1'
DNS_ZONE_DELEGATION_1_DOMAIN_1='firma.de'
```

Then, DNS queries for xx.firma.de are answered from the company's internal DNS server, otherwise the DNS server on the Internet is used

Another case:

- Circuit 1: Internet
- Circuit 2: Company network 192.168.1.0 *with* Internet-Access

Here we have two possibilities to reach the internet. To separate private from business, the following can be used:

```
ISDN_CIRC_1_ROUTE='0.0.0.0'
ISDN_CIRC_2_ROUTE='0.0.0.0'
```

We set a default route on both circuits and switch the route with the imond-client then - as desired. Also in this case set `DNS_ZONE_DELEGATION_N` and `DNS_ZONE_DELEGATION_x_DOMAIN_x` as described above.

If you want the reverse DNS resolution for such a network (e.g. an mail server will need this) you can provide the optional variable `DNS_ZONE_DELEGATION_x_NETWORK_x`, which lists the networks for active Reverse-Lookup. The following example illustrates this:

```
DNS_ZONE_DELEGATION_N='2'
    DNS_ZONE_DELEGATION_1_UPSTREAM_SERVER_N='1'
    DNS_ZONE_DELEGATION_1_UPSTREAM_SERVER_1_IP='192.168.1.12'
    DNS_ZONE_DELEGATION_1_DOMAIN_N='1'
    DNS_ZONE_DELEGATION_1_DOMAIN_1='firma.de'
    DNS_ZONE_DELEGATION_1_NETWORK_N='1'
    DNS_ZONE_DELEGATION_1_NETWORK_1='192.168.1.0/24'
    DNS_ZONE_DELEGATION_2_UPSTREAM_SERVER_N='1'
    DNS_ZONE_DELEGATION_2_UPSTREAM_SERVER_1_IP='192.168.2.12'
    DNS_ZONE_DELEGATION_2_DOMAIN_N='1'
    DNS_ZONE_DELEGATION_2_DOMAIN_1='bspfirma.de'
    DNS_ZONE_DELEGATION_2_NETWORK_N='2'
    DNS_ZONE_DELEGATION_2_NETWORK_1='192.168.2.0/24'
    DNS_ZONE_DELEGATION_2_NETWORK_2='192.168.3.0/24'
```

with the config option `DNS_ZONE_DELEGATION_x_UPTREAM_SERVER_x_QUERYSOURCEIP` you can define the source IP-address for outgoing DNS requests to upstream servers. This is useful i.e. if you reach the upstream DNS server via a VPN and and don't want the local VPN address of fli4l to appear as the source IP at the upstream server. Another usecase is an IP address not routable for the Upstream DNS server (could happen in a VPN). In this case it is as well necessary to set the IP address used by the dnsmasq to an IP used by fli4l to be accessible by the Upstream DNS Server.

```
DNS_ZONE_DELEGATION_N='1'
DNS_ZONE_DELEGATION_1_UPSTREAM_SERVER_N='1'
DNS_ZONE_DELEGATION_1_UPSTREAM_SERVER_1_IP='192.168.1.12'
DNS_ZONE_DELEGATION_1_UPSTREAM_SERVER_1_QUERYSOURCEIP='192.168.0.254'
DNS_ZONE_DELEGATION_1_DOMAIN_N='1'
DNS_ZONE_DELEGATION_1_DOMAIN_1='firma.de'
DNS_ZONE_DELEGATION_1_NETWORK_N='1'
DNS_ZONE_DELEGATION_1_NETWORK_1='192.168.1.0/24'
```

**DNS_REBINDOK_N DNS_REBINDOK_x_DOMAIN** The nameserver *dnsmasq* normally declines responses from other name servers containing IP addresses from private networks. It prevents a certain class of network attacks. But if you have a domain with private IP addresses and a separate name server that is responsible for this network, exactly the answers which would be rejected from *dnsmasq* are needed. List theese domains in `DNS_REBINDOK_x`, to accept answers from this domain.

Another example for nameservers delivering private IP-Addresses as an answer are so called "Real-Time Blacklist Server". An example based on these might look like this:

```
DNS_REBINDOK_N='8'
DNS_REBINDOK_1_DOMAIN='rfc-ignorant.org'
DNS_REBINDOK_2_DOMAIN='spamhaus.org'
DNS_REBINDOK_3_DOMAIN='ix.dnsbl.manitu.net'
DNS_REBINDOK_4_DOMAIN='multi.surbl.org'
DNS_REBINDOK_5_DOMAIN='list.dnswl.org'
DNS_REBINDOK_6_DOMAIN='bb.barracudacentral.org'
DNS_REBINDOK_7_DOMAIN='dnsbl.sorbs.net'
DNS_REBINDOK_8_DOMAIN='nospam.login-solutions.de'
```

### 1.1.3 DHCP-server

**OPT_DHCP** With `OPT_DHCP` you can activate the DHCP-server.

**DHCP_TYPE** (optional)

With this variable you can set if the internal DHCP-funktion of the dnsmasq should be used or if you want to use the external ISC-DHCPD. When using the ISC-DHCPD support for DDNS is not available.

**DHCP_VERBOSE** activates additional messages of DHCP in the log.

**DHCP_LS_TIME_DYN** determines the default lease-time for dynamically assigned IP-Addresses.

**DHCP_MAX_LS_TIME_DYN** determines the maximum lease-time for dynamically assigned IP-Addresses.

**DHCP_LS_TIME_FIX** Default lease-time for dynamically assigned IP-Addresses.

**DHCP_MAX_LS_TIME_FIX** determines the maximum lease-time for statically assigned IP-Addresses.

**DHCP_LEASES_DIR** Determines the folder for the leases-file. You may set an absolute path or *auto*. When *auto* is sued the lease file will be saved in a subdir of the persistent directory (see documentation for package base).

**DHCP_LEASES_VOLATILE** If the folder for the *Leases* resides inside the ram-disc (because the router boots i.e. from a CD or from another non-writeable device) the router will warn about a missing *Lease* file at each boot. This warning can be ommitted by setting `DHCP_LEASES_VOLATILE` to *yes*.

**DHCP_WINSSERVER_1** sets the address of the first WINS-Server. If the WINS server is configured and activated in the SAMBA package the settings from there are used.

**DHCP_WINSSERVER_2** sets the address of the second WINS-Server. If the WINS server is configured and activated in the SAMBA package the settings from there are used.

#### Local DHCP Ranges

**DHCP_RANGE_N** Number of DHCP ranges

**DHCP_RANGE_x_NET** Reference to one of the `IP_NET_x` networks

**DHCP_RANGE_x_START** sets the first IP-Address that can be used.

**DHCP_RANGE_x_END** sets the last assignable IP-Address. Both variables `DHCP_RANGE_x_START` and `DHCP_RANGE_x_END` could be left empty, so there will be no DHCP-Range, but hosts with MAC assignments will receive their values from the other variables.

**DHCP_RANGE_x_DNS_SERVER1** sets the addresse of the DNS-server for DHCP-hosts of the network. This variable is optional. If left empty or omitted, the IP-address of the matching network is used. Further it's possible to set this variable to 'none'. Then, no DNS-server is assigned.

**DHCP_RANGE_x_DNS_SERVER2** same settings for the second DNS-servers

**DHCP_RANGE_x_DNS_DOMAIN** sets a special DNS-domain for DHCP-hosts for this range. This variable is optional. If left empty or omitted, the default DNS-domain `DOMAIN_NAME` is used.

**DHCP_RANGE_x_NTP_SERVER** sets the address of the NTP-server for DHCP-hosts in this range. This variable is optional. If left empty or omitted, the IP-address of the `DHCP_RANGE_x_NET` network is used when a timeserver package is activated. If set to 'none', no NTP-Server is assigned.

**DHCP_RANGE_x_GATEWAY** sets the address of the gateway for this range. This variable is optional. If left empty or omitted, the IP-address of the `DHCP_RANGE_x_NET` network is used. If set to 'none', no gateway is assigned.

**DHCP_RANGE_x_MTU** sets the MTU for clients in this range. This variable is optional.

**DHCP_RANGE_x_OPTION_N** allows the setting of user defined options for this range. The available options are mentioned in the dnsmasq manual (http://thekelleys.org.uk/dnsmasq/docs/dnsmasq.conf.example). They are adopted unchecked - this could raise problems. This variable is optional.

### Extra DHCP-Range

**DHCP_EXTRA_RANGE_N** sets the number of DHCP-ranges not assigned to local networks. For this a DHCP-relay has to be installed on the gateway of the remote network.

**DHCP_EXTRA_RANGE_x_START** first IP-address to be assigned.

**DHCP_EXTRA_RANGE_x_END** last IP-address to be assigned.

**DHCP_EXTRA_RANGE_x_NETMASK** Netmask for this range.

**DHCP_EXTRA_RANGE_x_DNS_SERVER** Address of the DNS-servers for this range.

**DHCP_EXTRA_RANGE_x_NTP_SERVER** Address of the NTP-server for this range.

**DHCP_EXTRA_RANGE_x_GATEWAY** Address of the default-gateway for this range.

**DHCP_EXTRA_RANGE_x_MTU** MTU for clients in this range. This variable is optional.

**DHCP_EXTRA_RANGE_x_DEVICE** Network interface over which this range can be reached.

#### Not allowed DHCP-clients

**DHCP_DENY_MAC_N** Number of MAC-Addresses of hosts which should be rejeced.

**DHCP_DENY_MAC_x** MAC-Address of the host which should be rejeced.

#### Support For Network Booting

The dnsmasq supports clients booting by Bootp/PXE over the network. The needed informations for this are provided by dnsmasq and are configured per subnet and host. The needed variables are in the DHCP_RANGE_%- and HOST_%-sections and point to the bootfile (*_PXE_FILENAME), the server which hosts this file (*_PXE_SERVERNAME and *_PXE_SERVERIP) and perhaps necessary options (*_PXE_OPTIONS). Furthermore the internal tftp-server can be activated to provide network booting entirely from dnsmasq.

**HOST_x_PXE_FILENAME DHCP_RANGE_x_PXE_FILENAME** The bootfile. If PXE is used, the pxe-bootloader, i.e. pxegrub, pxelinux or similar.

**HOST_x_PXE_SERVERNAME HOST_x_PXE_SERVERIP DHCP_RANGE_x_PXE_SERVERNAME** Name and IP of the tftp-servers. If empty, the router itself is used.

**DHCP_RANGE_x_PXE_OPTIONS HOST_x_PXE_OPTIONS** Some bootloader need special options to boot. I.e. pxegrub asks by use of option 150 for the name of the menu file. This options can be put here. For pxegrub it looks like this:

```
HOST_x_PXE_OPTIONS='150,"(nd)/grub-menu.lst"'
```

If more options are needed, separate them by a space.

### 1.1.4 DHCP-Relay

A DHCP-relay is used, when another DHCP-Server manages the ranges which is not directly reachable from the clients.

**OPT_DHCPRELAY** Set to 'yes' to act as a DHCP-relay. To act as a DHCP-server is not allowed at the same time.

Default setting: `OPT_DHCPRELAY`='no'

**DHCPRELAY_SERVER** Insert the right DHCP-server to which the queries should be forwarded.

**DHCPRELAY_IF_N DHCPRELAY_IF_x** Sets `DHCPRELAY_IF_N` the number of network interfaces, the relay-server listens to. In `DHCPRELAY_IF_x` provide the apropriate network interfaces.

Provide the interface the DHCP-server uses to answer as well. Make sure to set the routes on the CP running the DHCP server are correct. The answer of the DHCP server is deirected to the interface to which the clioent is connected.

Assume the following szenario:

- relay with two interfaces
- interface to the clients: eth0, 192.168.6.1
- interface to the DHCP-server: eth1, 192.168.7.1
- DHCP-server: 192.168.7.2

A route on the DHCP server has to exist over which the answers to 192.168.6.1 can reach their destination. If the router on which the relay is running is the default gateway for the DHCP server everything os fine already. If not, an extra route is needed. If the DHCP server is a fli4l ythe following config variable is sufficient: IP_ROUTE_x='192.168.6.0/24 192.168.7.1'

There may be warnings about ignoring certain packets which you may safely ignore.

Example:

```
OPT_DHCPRELAY='yes'
DHCPRELAY_SERVER='192.168.7.2'
DHCPRELAY_IF_N='2'
DHCPRELAY_IF_1='eth0'
DHCPRELAY_IF_2='eth1'
```

### 1.1.5 TFTP-server

To deliver files with the TFTP-protocol, a TFTP-server is needed. This may be useful for netboot scenarios.

**OPT_TFTP** Activates the internal TFTP-server of the dnsmasq. Default setting is 'no'.

**TFTP_PATH** Specifies the folder, where the files to be delivered to the clients are stored. The files have to be stored manually there.

### 1.1.6 YADIFA - Slave DNS Server

**OPT_YADIFA** Activates the YADIFA Slave DNS Server. Default Setting: 'no'.

**OPT_YADIFA_USE_DNSMASQ_ZONE_DELEGATION** If this setting is activated the yadifa start script will automatically generate the according zone delegation entries for dnsmasq. The slave zones can be queried directly from dnsmasq and basically YADIFA_LISTEN_x entries not needed. Queries will only be forwarded to yadifa which is listening on localhost:35353 and then answered by dnsmasq.

**YADIFA_LISTEN_N** If you specified `OPT_YADIFA`='yes' you may provide local IPs on which yadifa is allowed to answer queries. `YADIFA_LISTEN_N` sets the number and `YADIFA_LISTEN_1` to `YADIFA_LISTEN_N` the local IPs. A port number is otional, with 192.168.1.1:5353 teh YADIFA Slave DNS Server would listen to DNS queries on port 5353. Please note that dnsmasq is not allowed to listen on all interfaces in this case (see `DNS_BIND_INTERFACES`). Only IPs of existing interfaces may be used here (ethernet, wlan ...) otherwise there will be warning during router boot. As an alternative it is possible to use an ALIAS name, like i.e. `IP_NET_1_IPADDR`

**YADIFA_ALLOW_QUERY_N**

**YADIFA_ALLOW_QUERY_x** Sets the IP addresses and nets that are allowed to access YADIFA. This setting will be used by YADIFA to configure fli4l's packet filter accordingly and to generate the configuration files for YADIFA. By the prefix "!" acces to YADIFA is denied for the IP address or network in question.

The fli4l packet filter will be configured in a way that all nets allowed in this variable and those for the zones are joined in an ipset list (yadifa-allow-query). A differentiation on zones is not possible for the packet filter. In addition all IP addressesand nets from this global setting whose access is denied will be added to the list. So you can't reenable access later on.

**YADIFA_SLAVE_ZONE_N** Specifies the number of slave DNS zones YADIFA should take care of.

**YADIFA_SLAVE_ZONE_x** The name of the slave DNS zone.

**OPT_YADIFA_SLAVE_ZONE_USE_DNSMASQ_ZONE_DELEGATION** Activates (='yes') or deactivates (='no') the dnsmasq zone delegation only for the slave zone.

**YADIFA_SLAVE_ZONE_x_MASTER** The IP address of the DNS master server with an optional port number.

**YADIFA_SLAVE_ZONE_x_ALLOW_QUERY_N**

**YADIFA_SLAVE_ZONE_x_ALLOW_QUERY_x** Specifies IP addresses and nets for which access to this YADIFA DNS zone is allowed. This can be used to limit access to certain DNS zones even more. YADIFA uses this setting to generate its configuration files.

By the prefix "!" acces to YADIFA is denied for the IP address or network in question.

# List of Figures

# List of Tables

# Index