

# DHCP and DNS Cache

## Install dnsmasq

I decided to use [dnsmasq](#) since it can fulfill multiple roles as both a DHCP and DNS cache. I'll first configure it for IPv4 and then later add in the few extra IPv6 lines needed.

## Setup DHCP

The following can look complicated but that is just because there are a ton of [MAC Addresses](#) and [IP Addresses](#) mixed throughout. If you look closely you can see that there are only four types of lines.

1. `no-dhcp-interface=eth0,lo` prevents DHCP binding on our loopback address and `eth0` which is the interface facing the Internet.
2. `dhcp-range=` declares a start and stop address and lease lifetime for each subnet. I am also setting an optional tag for each so I can target them later if I want.
3. `dhcp-option=` allows me to set specific DHCP options. The `tag:` allows me to target addresses matching a specific tag. I am overriding the default DNS servers because I want `lan` and `dmz` to use my *Pi-hole* server and `warp` should use a public DNS server since any device on that subnet is routed through a VPN tunnel so it doesn't have local network access.
4. `dhcp-host=` defines what IP addresses and hostnames get assigned to which network device with a specific MAC address

```
# /etc/dnsmasq.d/dhcp.conf
+ no-dhcp-interface=eth0,lo
+
+ dhcp-range=set:lan,10.0.5.1,10.0.7.254,12h
+ dhcp-range=set:dmz,10.0.8.1,10.0.8.254,12h
+ dhcp-range=set:warp,10.0.9.1,10.0.9.254,5m
+
+ dhcp-option=tag:lan,option:dns-server,10.0.1.2
+ dhcp-option=tag:lan,option:dns-server,10.0.1.2
+ dhcp-option=tag:warp,option:dns-server,1.1.1.1,1.0.0.1
```

```

+
+ # LAN - network infrastructure
+ dhcp-host=aa:af:57:f3:4e:90,10.0.1.2,pihole[]# pihole
+ dhcp-host=b4:fb:e4:8f:f9:74,10.0.1.3,unifi-switch-8[]# unifi-switch-8
+
+ # LAN - proxmox
+ dhcp-host=e0:d5:5e:63:fe:30,10.0.3.2,blackbox[]# blackbox
+ dhcp-host=70:85:c2:fe:4c:b7,10.0.3.3,mini[]# mini
+ dhcp-host=6e:91:84:4a:74:f1,10.0.3.4,backup[]# backup
+
+ # LAN - assigned devices
+ dhcp-host=d0:a6:37:ed:8c:7f,10.0.4.4,silverbook[]# silverbook
+ dhcp-host=82:13:00:9c:c7:00,10.0.4.5,thunderbolt[]# thunderbolt
+ dhcp-host=34:36:3b:7f:18:1e,10.0.4.8,jess[]# jess
+ dhcp-host=96:64:5f:1c:a6:2c,10.0.5.6,refuge[]# refuge
+ dhcp-host=7a:bc:46:d1:a3:1b,10.0.5.9,unifi[]# unifi
+
+ # DMZ - assigned devices
+ dhcp-host=62:59:92:a7:1d:f1,10.0.8.5,bitcoin[]# bitcoin
+ dhcp-host=32:cc:fb:a3:1a:57,10.0.8.2,contained[]# contained

```

## Setup DNS Caching

Everything here is commented with an explanation of what it does. The only thing slightly interesting is I have two `server=` parameters pointing to the IPv4 loopback addresses which is where *Unbound* is listening. If *Unbound* wasn't being used I'd either remove `no-resolv` and use the system nameservers or change the `server=` parameters to point to a [public recursive name sever](#).

```

# /etc/dnsmasq.d/dns.conf
+ # Add the domain to simple names (without a period) in /etc/hosts in the same way as for DHCP-derived
names.
+ expand-hosts
+
+ # Log the results of DNS queries handled by dnsmasq.
+ log-queries
+
+ # Do not listen on the specified interface.
+ except-interface=eth0,lo
+

```

```
+ # Accept DNS queries only from hosts whose address is on a local subnet, ie a subnet for which an interface
exists on the server.
+ local-service
+
+ # Dnsmasq binds the address of individual interfaces, allowing multiple dnsmasq instances, but if new
interfaces or addresses appear, it automatically listens on those
+ bind-dynamic
+
+ # Return answers to DNS queries from /etc/hosts and --interface-name which depend on the interface over
which the query was received.
+ localise-queries
+
+ # All reverse lookups for private IP ranges (ie 192.168.x.x, etc) which are not found in /etc/hosts or the DHCP
leases file are answered with "no such domain"
+ bogus-priv
+
+ # Later versions of windows make periodic DNS requests which don't get sensible answers from the public
DNS and can cause problems by triggering dial-on-demand links.
+ filterwin2k
+
+ # Enable code to detect DNS forwarding loops
+ dns-loop-detect
+
+ # Reject (and log) addresses from upstream nameservers which are in the private ranges.
+ stop-dns-rebind
+
+ # Exempt 127.0.0.0/8 and ::1 from rebinding checks.
+ rebind-localhost-ok
+
+ # Tells dnsmasq to never forward A or AAAA queries for plain names, without dots or domain parts, to
upstream nameservers.
+ domain-needed
+
+ # Specifies DNS domains for the DHCP server.
+ domain=hermz.io
+
+ # Don't read /etc/resolv.conf. Get upstream servers only from the command line or the dnsmasq configuration
file.
+ no-resolv
+
```

```
+ server=127.0.0.1
+ server=::1
```

# Resolve Static Clients

One problem I ran into was that static clients never use DHCP so the DHCP server doesn't register their hostname with their intended IP address. To work around this limitation I just added those entries to the `/etc/hosts` file since by default *dnsmasq* will resolve using those entries too.

```
# /etc/hosts
+ 10.0.1.1    ember
+ 10.0.1.2    pihol
+ 10.0.1.3    unifi-switch-8
+ 10.0.3.2    blackbox
+ 10.0.3.3    mini
+ 10.0.3.4    backup
+ 10.0.3.5    edge

# --- BEGIN PVE ---
```

# Reboot

Now that *dnsmasq* is fully configured I just restart it using *systemctl*

```
# systemctl restart dnsmasq.service
```

---

Revision #9

Created 31 March 2021 21:01:46 by dustin@swigg.net

Updated 8 April 2021 12:53:03 by dustin@swigg.net