# ARPD Daemon

Alexey Kuznetsov, kuznet@ms2.inr.ac.ru some_negative_number, 20 Sep 2001

arpd is daemon collecting gratuitous ARP information, saving it on local disk and feeding it to kernel on demand to avoid redundant broadcasting due to limited size of kernel ARP cache. **Description**

The format of the command is:

```
arpd OPTIONS [ INTERFACE [ INTERFACE ... ] ]
```

`OPTIONS` are:

- `-l` - dump `arpd` database to stdout and exit. Output consists of three columns: interface index, IP address and MAC address. Negative entries for dead hosts are also shown, in this case MAC address is replaced by word `FAILED` followed by colon and time when the fact that host is dead was proven the last time.
- `-f FILE` - read and load `arpd` database from `FILE` in text format similar dumped by option `-l`. Exit after load, probably listing resulting database, if option `-l` is also given. If `FILE` is `-`, `stdin` is read to get ARP table.
- `-b DATABASE` - location of database file. Default location is `/var/lib/arpd/arpd.db`.
- `-a NUMBER` - `arpd` not only passively listens ARP on wire, but also send brodcast queries itself. `NUMBER` is number of such queries to make before destination is considered as dead. When `arpd` is started as kernel helper (i.e. with `app_solicit` enabled in `sysctl` or even with option `-k`) without this option and still did not learn enough information, you can observe 1 second gaps in service. Not fatal, but not good.
- `-k` - suppress sending broadcast queries by kernel. It takes sense together with option `-a`.
- `-n TIME` - timeout of negative cache. When resolution fails `arpd` suppresses further attempts to resolve for this period. It makes sense only together with option `-k`. This timeout should not be too much longer than boot time of a typical host not supporting gratuitous ARP. Default value is 60 seconds.
- `-R RATE` - maximal steady rate of broadcasts sent by `arpd` in packets per second. Default value is 1.
- `-B NUMBER` - number of broadcasts sent by `arpd` back to back. Default value is 3. Together with option `-R` this option allows to police broadcasting not to exceed `B+R*T` over any interval of time `T`.

`INTERFACE` is name of networking inteface to watch. If no interfaces given, `arpd` monitors all the interfaces. In this case `arpd` does not adjust `sysctl` parameters, it is supposed user does this himself after `arpd` is started.

Signals

`arpd` exits gracefully syncing database and restoring adjusted `sysctl` parameters, when receives `SIGINT` or `SIGTERM`. `SIGHUP` syncs database to disk. `SIGUSR1` sends some statistics to `syslog`. Effect of another signals is undefined, they may corrupt database and leave `sysctl` parameters in an unpredictable state.

Note

In order to `arpd` be able to serve as ARP resolver, kernel must be compiled with the option `CONFIG_ARPD` and, in the case when interface list is not given on command line, variable `app_solicit` on interfaces of interest should be set in `/proc/sys/net/ipv4/neigh/*`. If this is not made `arpd` still collects gratuitous ARP information in its database.

Examples

1. Start `arpd` to collect gratuitous ARP, but not messing with kernel functionality:

   ```
   arpd -b /var/tmp/arpd.db
   ```

2. Look at result after some time:

   ```
   killall arpd
   arpd -l -b /var/tmp/arpd.db
   ```

3. To enable kernel helper, leaving leading role to kernel:

   ```
   arpd -b /var/tmp/arpd.db -a 1 eth0 eth1
   ```

4. Completely replace kernel resolution on interfaces `eth0` and `eth1`. In this case kernel still does unicast probing to validate entries, but all the broadcast activity is suppressed and made under authority of `arpd`:

   ```
   arpd -b /var/tmp/arpd.db -a 3 -k eth0 eth1
   ```

   This is mode which `arpd` is supposed to work normally. It is not default just to prevent occasional enabling of too aggressive mode occasionally.