User Manual

This page contain user manual for PepSAL.

By default, PEPsal will be launched as a service, running the pepsal binary as a daemon.

Parameters

PEPsal binary can be run with the following optional parameters:

- `-d` (`daemon`): run the binary in the background.
- `-v` (`verbose`): allow debug logs.
- `-h` (`help`): print the usage, and exit.
- `-f` (`fastopen`): enable using TCP FastOpen with the PEP sockets (must also be enabled at OS level).
- `-p` (`port`): the PEP listening port (by default, `5000`).
- `-V` (`version`): print version, and exit.
- `-a` (`ip_address`): address to bind the listening port (by default, `0.0.0.0`, all the interfaces).
- `-l` (`log_file`): file to log the active connections periodically.
- `-g` (`gcc_interval`): connections garbage collector that removes no longer entries from hash tables. (by default, 15 hours)
- `-t` (`pending_lifetime`): maximum lifetime for a pending connection. Past this time, it will be considered garbage. (by default, 5 hours)
- `-c` (`max_conns`): maximum number of connections allowed (by default, 2112).

In order to configure these parameters for the PEPsal service, the file `/etc/pepsal/pepsal.conf` must be modified, and the service restarted. This file contains variables for all the parameters already described.

iptables

Besides from running the binary, the traffic to optimize must be redirected to the PEPsal interface. This redirection is done by `netfilter`, and can be configured by using the tool `iptables`.

The target to use to redirect traffic to PEPsal is TPROXY, which can be used only in the PREROUTING chain (that treats packets before being routed) of the `mangle` table. This target receives two parameters: the port to which redirect the traffic (the port of PEPsal), and a mark to set to the packets. This mark will be used to route the packets.

As filter, any rule can be used: incoming interface, source address and/or port, destination address and/or port. In any case, since only TCP traffic is optimized, it is recommended to specify it to avoid unnecessary processing.

For example, to filter all incoming traffic on the interface `eth0`, the command to add this rule is:

```
iptables -A PREROUTING -t mangle -p tcp -i eth0 -j TPROXY --on-port 5000 --
tproxy-mark 1
```

To effectively direct these packets to PEPsal, they have to be routed to the local `lo` interface. A default route can be added for packets with a `fwmark`, so that the routing of other packets is not disturbed. To instruct the kernel to use a particular routing table (not the default) for marked packets, run the following command:

```
ip rule add fwmark 1 lookup 100
```

This commands tells the kernel to use the routing table number `100`, when routing packets with a forwarding mark equal to `1`.

Finally, a default route must be added to this new table, telling the kernel to route all packets to the loopback interface:

```
ip route add local 0.0.0.0/0 dev lo table 100
```

These two commands must be configured only one time, since they are valable for all PEPsal traffic. On the other hand, any number of iptables rules can be added, to filter any type of traffic.

In order to facilitate the iptables configuration, another binary is distributed with pepsal, `pepsal-iptables`, that allows to easily set iptables rules, along with the pertinent ip routes.