MiniUPnPd with PCP support

Peter Tatrai (ptatrai@cisco.com)

Reinaldo Penno (repenno@cisco.com)

Introduction

- MiniUPnPd
  - UPnP IGD and NATPMP daemon
  - used to control port mapping on Home gateways (e.g. OpenWRT, TomatoUSB and others)
  - Project’s website http://miniupnp.free.fr/
  - GitHub repository: https://github.com/miniupnp/miniupnp
- Goal:
  - Add PCP MAP operation support
  - Optionally add PCP PEER operation support
  - Create test platform for future PCP extensions i.e. flow priority, metadata …
    - Implemented experimental DSCP marking of flows signaled by PCP PEER operation and FLOWPRIORITY PCP option.

MiniUPnPd with PCP support

- Implementation challenges and details
  - support both existing NATPMP and a new PCP on same port 5351
  - PCP / NATPMP branch decision based on 1st byte (version for both protocols)
    - 0 -> NATPMP
    - 1 -> PCP
  - original NATPMP code had both receiving and processing of the packet implemented in one function
    - split into two functions ReceiveNATPMPOrPCPPacket and ProcessIncomingNATPMPPacket
    - Create new function ProcessIncomingPCPPacket to handle PCP messages
  - MAP operation uses same port mapping functions as a NATPMP and UPnP => works on all platforms
  - PEER operation needs to implement new port mapping functions
    - implementation was done only on Linux/netfilter platform

MiniUPnPd with PCP support

- Current status:
  - libpcp/miniupnp(https://github.com/libpcp/miniupnp) forked from main branch miniupnp/miniupnp
  - merged back into upstream repository on 7/12/2013
  - Pull request and patch is available at url: https://github.com/miniupnp/miniupnp/pull/41
- Obtaining source code:
  - Source code is now available in upstream version using git:

```
git clone https://github.com/miniupnp/miniupnp.git
```

  - Sending of PCP messages is possible by PCP client available at https://github.com/libpcp/pcp

MiniUPnPd with PCP support

- Building and running instructions:
  - Create config.h with enabled PCP MAP operation only:
    - cd miniupnp/miniupnpd
    - ./genconfig.sh --pcp
  - For both PCP PEER and MAP operations enabled:
    - ./genconfig.sh --pcp-peer
  - Compile sources for Linux platform
    - make -f Makefile.linux
  - Install iptables rules
    - sudo netfilter/iptables_init.sh
  - Edit miniupnpd.conf file. Fill correct values into ext_if_name and listening_ip
  - Run daemon on foreground
    - sudo ./miniupnpd -f miniupnpd.conf -d

MiniUPnPd with PCP demonstration

Server console:

```
server:$sudo ./miniupnpd -f miniupnpd.conf –d
miniupnpd[31907]: system uptime is 5572747 seconds
miniupnpd[31907]: Starting NAT-PMP/PCP UPnP-IGD with external interface eth1
miniupnpd[31907]: HTTP listening on port 46156
miniupnpd[31907]: Listening for NAT-PMP/PCP traffic on port 5351
miniupnpd[31907]: PCP request received from 10.20.40.2:51365 60bytes
miniupnpd[31907]: PCP MAP: added mapping TCP 1234->10.20.40.2:1234 'PCP 1234 tcp‘
miniupnpd[32040]: PCP request received from 10.20.40.2:41573 80bytes
miniupnpd[32040]: PCP PEER: added mapping UDP 10.20.40.2:1234(1234)->8.8.0.0:9 'PCP 1234 udp'
Ctrl+C
```

```
server:$ sudo iptables -t nat -nL MINIUPNPD
Chain MINIUPNPD (2 references)
target     prot opt source               destination
DNAT       tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:1234 to:10.20.40.2:1234
```

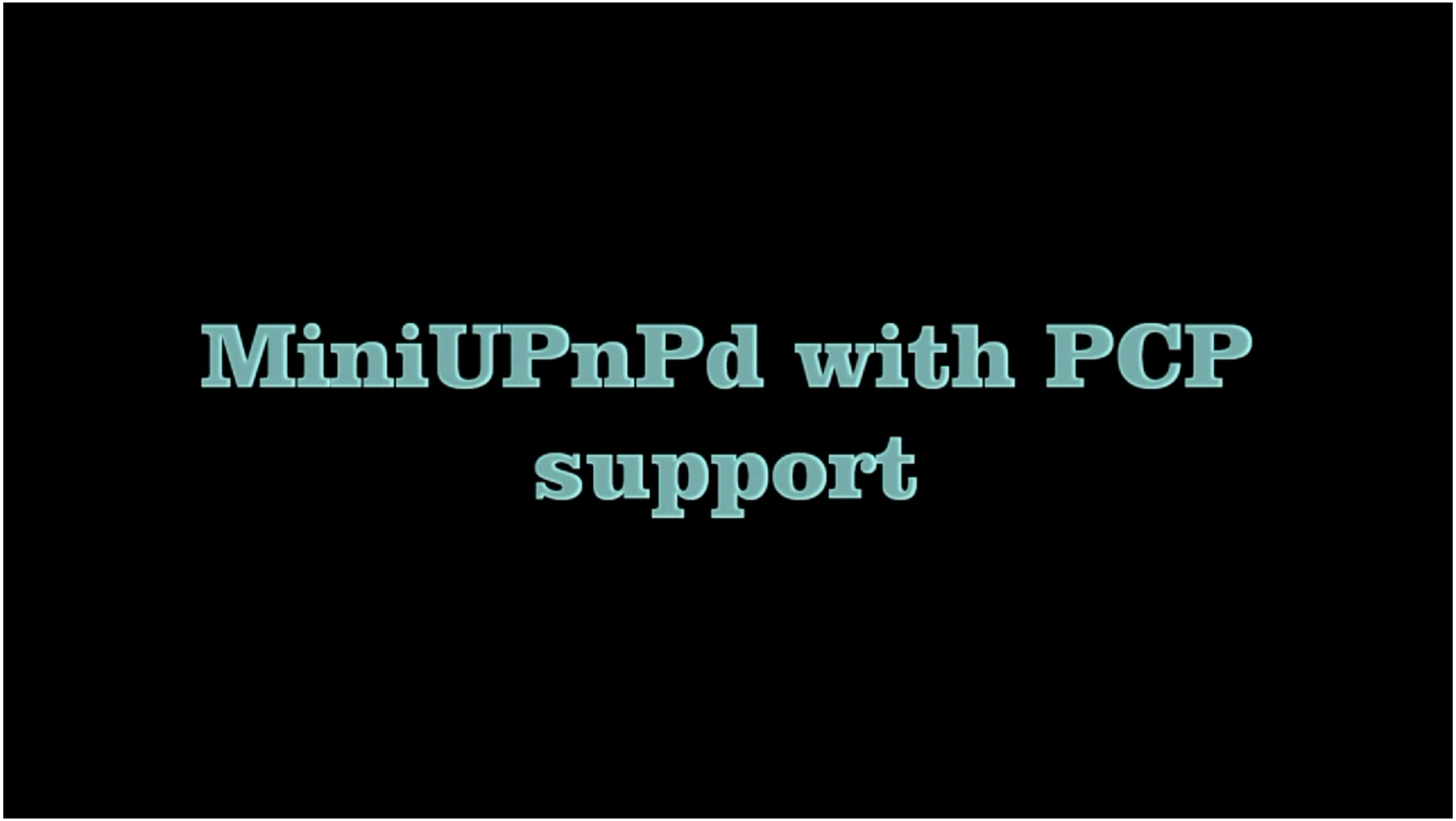
```
server:$ sudo iptables -t nat -nL MINIUPNPD-PCP-PEER
Chain MINIUPNPD-PCP-PEER (1 references)
target     prot opt source               destination
SNAT       udp  --  10.20.40.2           8.8.0.0              udp spt:1234 dpt:9 to:172.29.50.52:1234
```

Client console:

```
client:$ ./pcp -s <server_ip> -i :1234 -d
```

```
client:$ ./pcp -s <server_ip> -i :1234 -p 8.8.0.0:9 -u -d
```

MiniUPnPd with PCP demonstration



MiniUPnPd with PCP support