Proportional Differentiated Services Implementation for Linux

Benjamin Gaidioz
PhD student
Claude Bernard Univ. of Lyon 1/INRIA

Benjamin.Gaidioz[at]ens-lyon[dot]fr

Mathieu Goutelle
PhD student
LIP/INRIA

Mathieu.Goutelle[at]ens-lyon[dot]fr

Pascale Primet
Permanent Researcher
INRIA/LIP

Pascale.Primet[at]ens-lyon[dot]fr

This is the implementation of IP proportional differentiated services schedulers presented in Dovrolis99, as a Linux kernel module. The Linux PDS (Proportional Differentiated Services) queuing discipline implements both proportional delay and proportional loss rate IP differentiation. This file explains how to use it and describes its parameters.

Warning

This is a very pre-alpha release of the schedulers, only for testing purpose. The programs have been mainly and intensivly tested by ourselves, but this is the first distribution we make available to the public. Thus, especially the installation procedure is very "experimental". We request your feedback about all difficulties you have to face using this software and/or the documentation.

---

1. Copyright

PDS implementation - copyright 2002-2003 Benjamin Gaidioz, Pascale Primet and Mathieu Goutelle

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

2. Installation

Get the archive file `distrib_pds.tar.gz` from our web site (http://www.ens-lyon.fr/~mgoutell/distrib_pds/distrib_pds.tar.gz). Once you may downloaded it, uncompress it (it is rather small). Its contents is the following:

- `include.patch` and `net_sched.patch` which are both patches to the 2.4.18 version of the Linux kernel (see below for how to apply the patches),

- the `iproute/` directory which contains the entire source of the iproute package (the package that contains the **tc** command to control quality of service configuration),

- This `README` file, in various formats.

To install the queuing discipline, you must apply the two patches at the root of the directory containing the Linux kernel sources and compile the **tc** tool (which patched source code is in the `iproute` directory). The patches distributed in this version are for 2.4.18 kernel only. Suppose the kernel source tree is located in LINUX_SRC and the PDS patches are located in PDS_SRC.

1. Enter in LINUX_SRC and type:

   a. **patch -p0 PDS_SRC/include.patch**. This modifies some files in the LINUX_SRC/`include` directory (mainly adds PDS related fields to the struct sk_buff).

   b. **patch -p0 PDS_SRC/net_sched.patch**. This modifies some files in the LINUX_SRC/`net/sched` directory (the one that contains queuing disciplines) and created the `sch_pds.c` module.

2. Stay in LINUX_SRC.

      a.   Run the kernel configuration and tick the right option in the QoS section to enable the PDS scheduler as a module,

      b.   Save and exit the configuration,

      c.   As requested, run **`make dep`**

      d.   **`make modules`**

      e.   **`make modules_install`**

This should run correctly so that you can insert the newly compiled module by running **`modprobe sch_pds`**. You should see the module name in the **lsmod** output.

3. Go in the PDS_SRC/`iproute` directory.

      a.   Run **make** to compile the `tc` executable.

      b.   Install it in a directory included in your PATH.

3. Usage

Set up your first PDS queuing discipline this way using the appropriate **tc** command line. The syntax of the command to set up PDS is

tc qdisc add *parent* dev *nic* pds limit *qsize* nb *ncls* hist_depth *hdepth* fact_delay *dcoeff list* fact_loss *lcoeff list*

The `parent` is the name of the parent queuing discipline (`root` if PDS is the root queuing discipline), `nic` is the name of the network interface where the scheduler will be put on (say `eth0`). The `qsize` value correspond to the queue capacity (in packets). The `hdepth` value is the number of packets remembered in the history used to compute the loss rate. The `ncls` value is the number of classes in the scheduler. Be sure to set this value before declaring the coefficients for the differenciation (`dcoeff list` and `lcoeff list`).

• Delete all the already configured queuing discipline (be sure to replace the `eth0` device with the correct one, depending on your configuration):

```
tc qdisc del root dev eth0
```

If nothing is already configured, you get an error message (ignore it):

```
RTNETLINK answers: No such file or directory
```

• To set up the PDS scheduler with three classes and a specific configuration of delay and loss differentiation, use this command:

```
tc qdisc add root dev eth0 pds \
limit 200 hist_depth 1000 nb 3 \
fact_delay 1 2 3 fact_loss 3 2 1
```

You are now ready to use your PDS schedulers. To perform some tests, use a traffic generator, like MGEN, to create UDP streams and see if differentiation is effective (in delay, loss or both). Most of the tools permit the user to set the TOS field of the flow. The class ID (which ranges from 1 to `ncls`) is extracted from this field by the queuing discipline in comformance to RFC2474 (first six bits). You may read RFC2474 to ensure your marking is fine.

References

[Dovrolis, 1999] Constantinos Dovrolis and Parameswaran Ramanathan, *A case for relative differentiated services and the proportional differentiation model*, IEEE, 5, 13, 26-34, sep. 1999.

[PDS implementation, 2002] Benjamin Gaidioz, Mathieu Goutelle, and Pascale Primet, *Implementation of IP Proportional Differentiation with Waiting-Time Priority and Proportional Loss Rate dropper in Linux*, INRIA,   RR-4511 (http://www.inria.fr/rrrt/rr-4511.html) , Aug. 2002.

[RFC2474, 1998] Kathleen Nichols, Steven Blake, Fred Baker, and David Black, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*, IETF, dec. 1998.