

**NAME**

HFSC - Hierarchical Fair Service Curve's control under linux

**SYNOPSIS**

```
tc qdisc add ... hfsc [ default CLASSID ]
```

```
tc class add ... hfsc [ [ rt SC ] [ ls SC ] | [ sc SC ] ] [ ul SC ]
```

**rt** : realtime service curve

**ls** : linkshare service curve

**sc** : rt+ls service curve

**ul** : upperlimit service curve

- at least one of **rt**, **ls** or **sc** must be specified
- **ul** can only be specified with **ls** or **sc**

```
SC := [ [ m1 BPS ] d SEC ] m2 BPS
```

**m1** : slope of the first segment

**d** : x-coordinate of intersection

**m2** : slope of the second segment

```
SC := [ [ umax BYTE ] dmax SEC ] rate BPS
```

**umax** : maximum unit of work

**dmax** : maximum delay

**rate** : rate

For description of BYTE, BPS and SEC - please see **UNITS** section of [tc\(8\)](#).

**DESCRIPTION (qdisc)**

HFSC qdisc has only one optional parameter - **default**. CLASSID specifies the minor part of the default classid, where packets not classified by other means (e.g. u32 filter, CLASSIFY target of iptables) will be enqueued. If **default** is not specified, unclassified packets will be dropped.

**DESCRIPTION (class)**

HFSC class is used to create a class hierarchy for HFSC scheduler. For explanation of the algorithm, and the meaning behind **rt**, **ls**, **sc** and **ul** service curves - please refer to [tc-hfsc\(7\)](#).

As you can see in **SYNOPSIS**, service curve (SC) can be specified in two ways. Either as maximum delay for certain amount of work, or as a bandwidth assigned for certain amount of time. Obviously, **m1** is simply **umax/dmax**.

Both **m2** and **rate** are mandatory. If you omit other parameters, you will specify linear service curve.

**SEE ALSO**

[tc\(8\)](#), [tc-hfsc\(7\)](#), [tc-stab\(8\)](#)

Please direct bugreports and patches to: <netdev@vger.kernel.org>

**AUTHOR**

Manpage created by Michal Soltys (soltys@ziu.info)