

Table of Contents

Полисинг Общего канала 3

Полисинг Общего канала

Изменение настроек или отключение опции осуществляется с помощью редактирования файла конфигурации **/etc/dpi/fastdpi.conf**.



Для применения настроек необходимо выполнить релоад сервиса: **service fastdpi reload**

Для каждого из 8 доступных классов (групп) протоколов задаются ограничение на доступную им полосу.

При использовании **TBF (Token Bucket Function)** задается конкретная скорость для каждого класса трафика:

```
#Для входящего трафика
tbf_inbound_class0=rate 15mbit
tbf_inbound_class1=rate 15mbit
tbf_inbound_class2=rate 15mbit
tbf_inbound_class3=rate 15mbit
tbf_inbound_class4=rate 15mbit
tbf_inbound_class5=rate 15mbit
tbf_inbound_class6=rate 15mbit
tbf_inbound_class7=rate 15mbit
#Для исходящего трафика
tbf_class0=rate 15mbit
tbf_class1=rate 15mbit
tbf_class2=rate 15mbit
tbf_class3=rate 15mbit
tbf_class4=rate 15mbit
tbf_class5=rate 15mbit
tbf_class6=rate 15mbit
tbf_class7=rate 15mbit
```

Опционально можно указать параметры peakrate, burst, cburst. Если ограничение для какого-то класса не указано, то полоса для него не ограничена.

При использовании **HTB (Hierarchical Token Bucket)** задается общая скорость и скорость классов с заимствованием:

```
#Для входящего трафика
htb_inbound_root=rate 900mbit
htb_inbound_class0=rate 300mbit ceil 900mbit
htb_inbound_class1=rate 100mbit ceil 200mbit
htb_inbound_class2=rate 100mbit ceil 200mbit
htb_inbound_class3=rate 100mbit ceil 200mbit
htb_inbound_class4=rate 100mbit ceil 100mbit
htb_inbound_class5=rate 100mbit ceil 100mbit
htb_inbound_class6=rate 50mbit ceil 100mbit
```

```
htb_inbound_class7=rate 50mbit ceil 100mbit
#Для исходящего трафика
htb_root=rate 900mbit
htb_class0=rate 300mbit ceil 900mbit
htb_class1=rate 100mbit ceil 200mbit
htb_class2=rate 100mbit ceil 200mbit
htb_class3=rate 100mbit ceil 200mbit
htb_class4=rate 100mbit ceil 100mbit
htb_class5=rate 100mbit ceil 100mbit
htb_class6=rate 50mbit ceil 100mbit
htb_class7=rate 50mbit ceil 100mbit
```

htb_root - корневой класс, в котором указан общий размер полосы, в рамках которого проводится перераспределение и до которого можно расти если ceil не указан

rate - минимальный размер полосы

ceil - максимальный размер полосы, который можно заимствовать из корневого класса если полоса свободна

Опционально можно указать параметры burst, cburst. Сумма rate классов не должна превышать общий размер полосы, иначе поведение не определено.

Выделение полосы для некоторых классов (например пиринга) можно вывести из HTB иерархии, если в описании указать для них ключевое слово static, в этом случае ограничение для данного класса будет работать подобно tbf без привязки к htb_root

```
htb_inbound_class6=rate 200mbit static
htb_class6=rate 200mbit static
```

Для продвинутых пользователей: дополнительно можно включить механизм обратной связи, который работает следующим образом:

В настройках задается ограничение на входящий трафик, а для класса htb_root задается параметр ceil:

```
htb_inbound_root=rate 800mbit ceil 950mbit
htb_root=rate 450mbit ceil 600mbit
```

Когда входящий трафик превысит заданный rate=800mbit для исходящего трафика начинает уменьшаться верхняя граница ceil, заданная в параметре htb_root ceil=600mbit. Но даже при превышении входящим трафиком величины ceil=950mbit, исходящий трафик не будет ограничен больше, чем указано в параметре htb_root rate=450mbit. Процент превышения считается по интервалу ceil 950mbit ↔ rate 800mbit и на такой же процент сокращается исходящий трафик. При ограничении величины ceil, заданной в параметре htb_root, остальные классы начинают перераспределять трафик, чтобы не превысить общее ограничение установленной в htb_root.

Указанный механизм эффективно работает для протоколов построенных по принципу запрос-ответ, для которых при ограничении исходящего трафика (запрос) уменьшается количество входящего (ответ), а это большинство прикладных протоколов.

Примечания:

Если ограничение для класса не указано, то он может занимать всю доступную полосу. Соответствие между приоритетом dscr и классом для полисинга зависит от [настройки class_order](#)