

Table of Contents

2 Настройка	3
<i>Пример 1: Ограничение торрент</i>	<i>3</i>
<i>Пример 2: Максимальная скорость для пиринга</i>	<i>4</i>
<i>Пример 3: Назначение полисинга для мультипользователя</i>	<i>5</i>

2 Настройка

Полисинг имеет два варианта подключения:

1. Конфигурационный файл: подключается без указания профиля, создается индивидуальный профиль для каждого абонента.
2. Именованный профиль: подключается с указанием имени профиля.



Для BRAS необходимо использовать именованные профили **имя, которых указывается в атрибутах Radius-Accept.**

Для каждой из политик управления полосой создается конфигурационный файл, в котором задаются ограничения на доступную полосу в зависимости от класса (группы) протоколов (аналогично тому, как это сделано для [управления общей полосой](#))

Применение настроенных политик по отношению к абонентам осуществляется с помощью утилиты [fdpi_ctrl](#).

Формат команды:

```
fdpi_ctrl команда --policing файл_описания_полисинга [список_IP]
```

Подробнее синтаксис команд и способы задания IP адресов описаны в разделе [Команды управления](#)



Обратите внимание на использование [именованных профилей полисинга](#)
Также тарифный план можно задать в [формате JSON](#)

Пример 1: Ограничение торрент

Планируем предложить абонентам тарифный план на 10 Мбит/с, в котором торренты ограничены скоростью 3Мбит/с.

Для этого торренты должны быть выделены в отдельный класс, как описано в разделе [Назначение приоритетов](#).

```
default      cs0
bittorrent   cs1
```

В данном примере разделим трафик по протоколам всего на 2 класса, где

- cs0 - соответствует DSCP=0 QOS(IPP)=0 Best Effort
- cs1 - соответствует DSCP=8 QOS(IPP)=1 Priority

Создадим файл конфигурации `rateplan_1.cfg`, в котором для каждого из 8 доступных классов (групп) протоколов задаем ограничение на доступную им полосу. Используем HTB (дисциплина с заимствованием свободной полосы) и для торрентов указываем, что размер полосы ограничен 3mbit, но не меньше чем 1mbit. В отличии от торрентов прочий трафик может занимать всю доступную полосу.

```
htb_inbound_root=rate 10mbit
htb_inbound_class0=rate 8bit ceil 10mbit
htb_inbound_class1=rate 1mbit ceil 3mbit
htb_inbound_class2=rate 8bit ceil 10mbit
htb_inbound_class3=rate 8bit ceil 10mbit
htb_inbound_class4=rate 8bit ceil 10mbit
htb_inbound_class5=rate 8bit ceil 10mbit
htb_inbound_class6=rate 8bit ceil 10mbit
htb_inbound_class7=rate 8bit ceil 10mbit
htb_root=rate 10mbit
htb_class0=rate 8bit ceil 10mbit
htb_class1=rate 1mbit ceil 3mbit
htb_class2=rate 8bit ceil 10mbit
htb_class3=rate 8bit ceil 10mbit
htb_class4=rate 8bit ceil 10mbit
htb_class5=rate 8bit ceil 10mbit
htb_class6=rate 8bit ceil 10mbit
htb_class7=rate 8bit ceil 10mbit
```

- `htb_inbound_root`, `htb_root` - корневые классы, в которых указан общий размер полосы для входящего и исходящего трафика и в рамках которых проводится перераспределение полосы
- `rate` - минимальный размер полосы
- `ceil` - максимальный размер полосы, который можно заимствовать из корневого класса если полоса свободна
- `class2-7` фактически не будут использоваться, так как у нас в конфигурации определено только 2 класса 0 и 1



Сумма скорости по классам должна быть меньше, чем `root rate`.

Теперь абонентам с данным тарифом назначаем сконфигурированную политику:

```
fdpi_ctrl load --policing rateplan_1.cfg --file
subscribers_with_rateplan_1.txt
```

Пример 2: Максимальная скорость для пиринга

Выделение полосы для некоторых классов (например, содержащих пиринговый трафик) можно вывести из HTB иерархии, если в описании указать для них ключевое слово **static**. В этом случае ограничение для данного класса будет действовать независимо, без привязки к `htb_root`. Например, в примере выше зададим отдельное ограничение в 100 Мбит/с для класса

6.

```
htb_inbound_root=rate 10mbit
htb_inbound_class0=rate 8bit ceil 10mbit
htb_inbound_class1=rate 1mbit ceil 3mbit
htb_inbound_class2=rate 8bit ceil 10mbit
htb_inbound_class3=rate 8bit ceil 10mbit
htb_inbound_class4=rate 8bit ceil 10mbit
htb_inbound_class5=rate 8bit ceil 10mbit
htb_inbound_class6=rate 100mbit static
htb_inbound_class7=rate 8bit ceil 10mbit
htb_root=rate 10mbit
htb_class0=rate 8bit ceil 10mbit
htb_class1=rate 1mbit ceil 3mbit
htb_class2=rate 8bit ceil 10mbit
htb_class3=rate 8bit ceil 10mbit
htb_class4=rate 8bit ceil 10mbit
htb_class5=rate 8bit ceil 10mbit
htb_class6=rate 100mbit static
htb_class7=rate 8bit ceil 10mbit
```

Пример 3: Назначение полисинга для мультипользователя

Назначим тарифный план из примера 1 абоненту с несколькими IP. Проверяем, что в настройках `dpri` **`/etc/dpi/fastdpi.conf`** включена поддержка БД.

```
udr=1
```

Если не включена, то включаем и рестартуем `dpri`: `service fastdpi restart`

Закрепляем за корпоративным абонентом все его IP

```
fdpi_ctrl load --bind_multi --user
000_PizzaJohnes:192.168.0.1-192.168.0.5,192.168.1.10-192.168.1.25
```

Списком IP абонента можно управлять [динамически](#) (т.е. добавлять новые или удалять IP)

Назначаем ему ограничение полосы в соответствии с тарифным планом

```
fdpi_ctrl load --policing rateplan_1.cfg --login 000_PizzaJohnes
```



Для продвинутых пользователей: Рекомендуем ознакомиться с разделом ["Управление исходящим трафиком через обратную связь"](#).