

# Содержание

Резервирование BRAS .....	3
<i>Персистентная очередь</i> .....	3
<i>Применение данных на fastDPI</i> .....	4



# Резервирование BRAS

Вебинар по теме:



Video

В SKAT 8.3+ применяется следующая схема репликации для согласования данных об абонентах на всех fastDPI-серверах: fastPCRF шлет ответы авторизации и CoA-запросы на все серверы, перечисленные в параметрах [fdpi\\_server](#).



Отправка параметров авторизации производится через персистентную очередь, так что даже если какой-то из серверов fastDPI был отключен на момент отправки данных, при включении он получит все данные за время своего простоя.

## Персистентная очередь

Очередь организуется в файловой системе в каталоге `/var/spool/dpi/pcrf`. Для каждого fastDPI-сервера в этом каталоге создается отдельный файл с именем `pq-<fastDPI-address>:<port>`, например, если в `fastpcrf.conf` описаны два fastDPI-сервера:

```
fdpi_server=127.0.0.1%lo:29000
fdpi_server=10.20.30.40:eth1:29000
```

то каталог `/var/spool/dpi/pcrf` будет содержать два файла:

```
pq-127.0.0.1:29000
pq-10.20.30.40:29000
```

Можно изменить каталог очереди в `fastpcrf.conf` параметром `fdpi_pqueue_dir`:

```
# Каталог, в котором создаются персистентные очереди
```

```
fdpi_pqueue_dir=/var/spool/dpi/pcrf
```

Очередь страничная, размер одной страницы 2М. Можно задать максимальный размер очереди в страницах fastpcrf.conf-параметром `fdpi_pqueue_max_pagecount`:

```
# Max число страниц в очереди.  
# Размер каждой страницы = 2М  
# 0 - число страниц не ограничено.  
# Минимальное значение: 2  
# Следует учитывать, что размер файла очереди ограничен сверху параметром  
rlimit_fsize  
# больше этого размера очередь не может быть.  
fdpi_pqueue_max_pagecount=4
```

Если fastDPI-сервер online и его очередь пуста, данные шлются непосредственно серверу, минуя очередь. Если же сервер становится недоступным, данные записываются в его очередь. Если очередь переполняется, то есть невозможно распределить новую страницу, самая старая страница стирается из очереди, и на её месте создается новая страница; таким образом, очередь представляет собой циклический буфер.

Как только fastDPI-сервер становится доступным, ему посылаются все данные из его очереди. FastPCRF периодически пытается соединиться с отвалившимися fastDPI-серверами, тайм-аут этих попыток задается fastpcrf.conf-параметром `fdpi_reconnect_timeout`, по умолчанию 2 секунды:

```
# Тайм-аут попытки подсоединения к fastdpi-серверам, секунд  
#fdpi_reconnect_timeout=2
```

Можно вручную **при остановленном fastpcrf** удалять файлы очередей.

[CLI-команды управления очередями](#)

## Применение данных на fastDPI

При приеме данных авторизации сервер fastDPI видит, его это был запрос или же это ответ на чужой запрос (для этого в пакете есть специальная метка). Если это ответ на свой запрос, данные применяются "по полной": создается DHCP или PPPoE-сессия, если это был запрос DHCP или PPPoE-авторизации, данные запоминаются в UDR. Если же это ответ на чужой запрос, fastDPI просто запоминает "чужие" данные у себя в UDR. Тем самым при отключении основного fastDPI-сервера и переводе нагрузки на резервный, у резервного fastDPI-сервера в UDR уже будут все свойства абонента - его услуги, полисинг, L2-свойства - MAC-адрес, VLAN и пр. То есть UDR у основного и резервного серверов будут по большому счету согласованы.