

# Содержание

- BRAS reservation ..... 3
  - Persistent queue* ..... 3
  - Ways the data is applied on the DPI* ..... 4



# BRAS reservation

The following replication scheme is used in the Stingray Service Gateway ver. 8.3+ to align subscriber data on all fastDPI servers: fastPCRF sends authorization responses and CoA requests to all the servers listed in the [fdpi\\_server](#) configuration parameters.



Authorization parameters are sent by using a persistent queue so even if some of the fastDPI servers are inactive at the time of data transmission, they will receive all the data missed during their idle time once they are activated.

## Persistent queue

The queue is organized in the file system within the `/var/spool/dpi/pcrf` directory. An individual file is created for each fastDPI server in this directory using the `pq-<fastDPI-address>:<port>` name, for example, if there are two fastDPI servers described in `fastpcrf.conf`:

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

then `/var/spool/dpi/pcrf` directory will contain two following files:

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

You can change the queue directory in `fastpcrf.conf` by using the `fdpi_pqueue_dir` parameter:

```
# The directory where persistent queues are created
fdpi_pqueue_dir=/var/spool/dpi/pcrf
```

The queue is page-organized, the size of each page equals to 2M. The maximum queue size can be specified in pages through the `fdpi_pqueue_max_pagecount` `fastpcrf.conf` parameter:

```
# Max number of pages in the queue.
# Each page size = 2M
# 0 means that the number of pages is not limited.
# Minimum value: 2
# Please note that the size of queue file is limited from above by the
rlimit_fsize parameter
# There cannot be a queue larger than the size specified by
rlimit_fsize.
fdpi_pqueue_max_pagecount=4
```

If the fastDPI server is available and its queue is empty, the data will be sent directly to the server bypassing the queue. If the server becomes unavailable, the data will be written to its queue. If the queue is full, that is, it is impossible to allocate a new page, the oldest page will be deleted from the

queue, and a new one will be created in its place; thus, the queue is circular buffer.

Once the fastDPI server becomes available, all the data from its queue is sent to it. FastPCRF tries to connect to the unavailable fastDPI servers at a regular times, the timeout of these attempts is specified by the `fdpi_reconnect_timeout` `fastpcrf.conf` parameter, its default value is 2 seconds:

```
# Timeout attempting to connect to fastdpi servers, in seconds
#fdpi_reconnect_timeout=2
```

You can manually delete the queue files **when fastpcrf is stopped** .

[CLI-commands for managing the queues](#)

## Ways the data is applied on the DPI

When receiving authorization data, the fastDPI server identifies whether it was response to its own request or it was a response to another request (there is a special label in the packet for this purpose). If this is a response to its own request, the data will be applied completely: a DHCP or PPPoE session will be created in case of DHCP or PPPoE authorization request and the data will be stored in UDR. If this is an answer to another request, the fastDPI will simply store corresponding "extraneous" data in the UDR. Thus, when the main fastDPI server becomes unavailable all the load would be imposed on the backup fastDPI server and the latter will already have contain all the subscriber properties in its UDR: subscriber services, its policing, L2 properties - MAC address, VLAN, etc. That is, the UDR of the main and backup servers will essentially contain the consistent data.