

# Содержание

<b>Beta-версия 14.2</b> .....	3
<b><i>Изменения в версии 14.2 BETA8</i></b> .....	3
<b><i>Изменения в версии 14.2 BETA7</i></b> .....	3
<b><i>Изменения в версии 14.2 BETA6</i></b> .....	4
<b><i>Изменения в версии 14.2 BETA5</i></b> .....	6
<b><i>Изменения в версии 14.2 BETA4</i></b> .....	7
<b><i>Изменения в версии 14.2 BETA3</i></b> .....	8
<b><i>Изменения в версии 14.2 BETA2.1</i></b> .....	8
<b><i>Изменения в версии 14.2 BETA2</i></b> .....	8
<b><i>Изменения в версии 14.2 BETA1</i></b> .....	9



# Бета-версия 14.2

Отслеживайте текущую версию СКАТ и отправляйте заявки на обновление через [Личный кабинет](#)



Постоянная лицензия

СКАТ-10-COMPLETE

Версия ПО СКАТ: 13.2

[Доступно обновление 14.0](#)

## Изменения в версии 14.2 ВЕТА8

- [DPI] Изменено: после проверки IPSNI откат на базовый протокол или протокол, заданный по SNI (если определен)
- [DPI] Изменено: уменьшена глубина просмотра в попытках декодировать sname/sni
- [CLI] Добавлена команда записи rсар с порта:

```
dev rсар <dev-name> rx|tx|any|off
```

- o rx — записывать принятые с порта пакеты
- o tx — записывать отправляемые в порт пакеты
- o any — rx и tx
- o off — завершить запись

Префиксы файлов rсар (dev - имя порта):

- o rx-dev — для rx
- o tx-dev — для tx

- [NAT] Исправление консистентности очереди приватного адреса
- [DPI] Решение проблемы выбора tx-порта для многозначных конфигураций: обратный пакет приоритетно посылается в тот порт, из которого был первоначальный пакет
- [NAT] Исправление и оптимизация очереди портов приватного адреса:
  1. Очередь портов приватного адреса распределена по потокам
  2. Очередь портов приватного адреса разделена на "короткую" и "длинную".
- [CLI][RG] Добавлено: команда `rg show <IP>` просмотра текущих данных по рейтинг-группе для абонента

## Изменения в версии 14.2 ВЕТА7

- [DPI] Изменено: проверка на FakeSNI не производится если протокол определен по IP и нет mark1
- [CG-NAT] Оптимизация команды статистики `fdpi_ctrl list all status -service 11`
- [DPDK] Увеличен максимальный размер памяти до 256 ГБ
- [DPI][BRAS] Добавлена услуга 20: полисинг по рейтинг группам (RG) и контроль квот по

объему.

Создание профиля услуги 20:

1. Включить поддержку RG в `fastdpi.conf`  
`rating_group_count=0` — число `rating group`, 0 — RG отключены. Значение по умолчанию: 0
2. Подготовить текстовый файл, в котором для каждой рейтинг-группы прописать TBF-полисинг, квоту и действие по достижении квоты, пример:

```
rg4 tbf rate 1Mbit burst 1Mbit inbound.rate 8Mbit inbound.burst
1Mbit quota 100MB report
rg5 tbf rate 8Mbit burst 1Mbit inbound.rate 8Mbit inbound.burst
1Mbit quota 1GB block
```

`report` и `block` — это доступные действия по достижении квоты: `report` — сообщить о достижении квоты, но пропускать трафик далее; `block` — сообщить о достижении квоты и заблокировать трафик по данной рейтинг-группе

3. Конвертировать текстовый файл в бинарный формат:

```
cat rg.txt | lst2rg rg.bin
```

4. Поместить полученный бинарный файл в директорию, откуда его будет читать DPI:

```
cp rg.bin /var/lib/dpi/rg.bin
```

5. Создать профиль услуги:

```
fdpi_ctrl load profile --service 20 --profile.name rg1 --
profile.json '{ "rg_list" : "/var/lib/dpi/rg.bin" }'
```

`max_profiles_serv20` — настройка максимального количества профилей. По умолчанию — 32.

Утилита `rg2lst` позволяет декодировать бинарный файл в читаемый вид:

```
rg2lst rg.bin > rg.txt
```

5. [DPIUTILS] Добавлены утилиты `lst2rg` и `rg2lst` для конвертации профиля услуги 20

## Изменения в версии 14.2 BETA6

1. [DPI] Добавлена проверка `viber_cl` по контейнеру
2. [DPI] Исправлено: переопределение облачных протоколов некоторыми встроенными
3. [DPI] Исправлено: добавлено определение протокола адресам в случае когда SNI уже в первом пакете для соблюдения приоритета IP/SNI
4. [DPI] Исправлено: определение DSCP с первого пакета для облачных протоколов, заданных по адресам
5. [NAT] Добавлено явное закрытие TCP-соединения при переиспользовании порта другим абонентом
6. [CLI] Добавлены новые поля в `fdpi_cli dump flow cache command`. [Описание](#)
7. [NAT] Изменена работа с очередями публичных портов: порты с коротким временем

жизни и с длинным временем жизни живут в разных очередях. Порты теперь являются элементами подочереди приватного адреса. Порт, к которому обращались из non-owner потока, может переиспользоваться сразу

8. [CLI] Добавлена установка rating group и контроля tethering через 18 услугу, в настройке профиля которой добавлены новые опциональные поля:

tethN, где возможные варианты:

- teth0 — нет контроля tethering (default)
- teth1 — есть контроль tethering: tethering присутствует
- teth2 — есть контроль tethering: tethering отсутствует

rgN, где возможные варианты:

- rg0 default (rg не установлена)
- rg1 установлена rg=1
- ..
- rg65535 установлена rg=65535

### Пример задания 18 услуги:

1. подготавливаем текстовое описание настроек example.txt

```
http cs0 teth1 rg1
https cs0 teth1 rg1
http cs0 teth2 rg2
https cs0 teth2 rg2

dns cs1 teth1 rg1
dns cs1 teth2 rg2

default cs7 teth0 rg3
```



**В данном примере по http, https протоколам отслеживается тетеринг и в зависимости от этого назначается соответствующая rg. Отметим, что класс полисинга cs один и тот же. Аналогично по dns протоколу. По ВСЕМ остальным протоколам (default) нет контроля тетеринга и указана отдельная rg.**

2. конвертируем во внутренний формат

```
cat example.txt|lst2dscp /tmp/example.bin
```

3. опционально проверяем обратной конвертацией

```
dscp2lst /tmp/example.bin
```

4. создаем профиль 18 услуги и назначаем абоненту (либо сразу назначаем именованный профиль)

```
fdpi_ctrl load profile --service 18 --profile.name test_dscp
--profile.json '{ "dscp" : "/tmp/example.bin" }'
fdpi_ctrl load --service 18 --profile.name test_dscp --login
test_subs
```

Проверяем

```
fdpi_ctrl list --service 18 --login test_subs
```

В трассировке добавлено поле rg=N

9. [BRAS][DHCPv6] Исправлена отправка периодического ICMPv6 Router Adv DHCPv6-абонентам
10. [BRAS][pppoe] Исправлено изменение src/dst MAC в ethernet-заголовке пакета при терминации. Для PPPoE-пакетов терминация ethernet-заголовка должна проводиться всегда. Но при включенной настройке bras\_term\_by\_as=1 в ситуации, когда srcAS не помечена как term, изменения ethernet src/dst MAC не происходило.

## Изменения в версии 14.2 BETA5

1. [BASE] Добавлена поддержка LLDP.

При включении поддержки LLDP fastDPI начинает эмитировать LLDP-пакеты (LLDPDU) на указанных портах. Входящие LLDP-пакеты безусловно дропаются.

Новые параметры fastdpi.conf (все параметры являются горячими, задаются в секции `lldp`):

- `enable` — включение поддержки LLDP (булевый флаг). По умолчанию, если в файле конфигурации есть секция `lldp`, то `enable=on`; если такой секции нет, `enable=off`
- `chassis` — строка — значение Chassis-Id. TLV Chassis-Id является обязательным в LLDP-пакете. Если не задано — полагается равным MAC-адресу, заданному в опции `bras_arp_mac`, если этой опции нет — MAC-адресу порта
- `ttl` — число — значение TTL в секундах, по умолчанию 120
- `src_mac` — MAC-адрес — значение source MAC в ethernet-заголовке LLDPDU. Если не задано — полагается равным MAC-адресу, заданному в опции `bras_arp_mac`, если этой опции нет — MAC-адресу порта
- `dest_mac` — MAC-адрес — значение dest MAC в ethernet-заголовке LLDPDU. По умолчанию — 01:80:c2:00:00:0e (LLDPDU multicast)
- `system_name` — строка — значение TLV System-Name пакета LLDPDU. Если не задано — TLV System-Name не включается в LLDPDU.
- `system_desc` — строка — значение TLV System-Desc пакета LLDPDU. Если не задано — TLV System-Desc не включается в LLDPDU.
- `device=<имя_порта>;enable=<on|off>;desc=<port_desc>` — имена портов, для которых надо посылать LLDPDU. Каждый порт задается в отдельном параметре `device`, имя порта — то, что задано в `in_dev/out_dev`. Для каждого порта можно задать следующие опции:
  - `enable=on|off` — включить или отключить отправку LLDPDU для этого порта, по умолчанию `on` (включено)
  - `desc=строка` — значение TLV Port-Desc пакета LLDPDU; если не задано — TLV Port-Desc не включается в LLDPDU

Отладочные опции:

- `trace` — включение трассировки LLDP (булевый флаг)
  - `pcap` — запись LLDP-пакетов в PCAP (булевый флаг)
2. [CLI] Новые команды CLI: `lldp enable`, `lldp disable` — позволяют включать/выключать формирование LLDP-пакетов

3. [NAT] Улучшения в управлении лимитами сессий: для лимитов `nat_tcp_max_sessions/nat_udp_max_sessions`, задающих ограничения на количество выданных белых портов, исправлено уменьшение счетчика количества выданных портов, что могло приводить к небольшому превышению заданного лимита. Изменены счетчики `whpf`, `whp_salfs`, `whp_lalfs`, `whp_ruse`, `whp_ruse_salfs`, `whp_ruse_lalfs` и аналогичные счетчики в статистике по потокам (`thr_salfs` и другие), а также вывод команды `nat show`, чтобы отражать текущее актуальное использование портов, а не кумулятивное (накопительное)
4. [NAT] Исправлено: добавлена проверка актуальности NAT трансляций в режиме FullCone в случае `nat_whp_lifetime < lifetime_flow`: если в сессии появилась активность, а NAT-порт уже переиспользован, то выделяется новый порт

## Изменения в версии 14.2 BETA4

1. [DPI] Добавлено детектирование протокола FakeTLS с валидацией
2. [DPI] Исправлено: переключение с QUIC\_UNKNOWN на QUIC при успешном разборе SNI
3. [DHCP6-Proxy] Добавлена DHCPv6-опция 79 Client-LinkLayer-Address, содержащая MAC-адрес абонента, в Relay-Forward запросы к DHCPv6-серверу Framed-Pool
4. [VLAN-Rule][PPPoE] В команду `vlan rule show` добавлен вывод всех разрешений для Service-Name
5. [VLAN-Rule][PPPoE] Добавлено: полная поддержка Service-Name для QinQ. Поддерживаются правила:
  1. без селективности по CVLAN: правила вида `SVLAN.*` с указанием SName и без
  2. полный QinQ (`svan.cvlan`) с селективностью по SName
6. [VLAN-Rule][PPPoE] Рефакторинг поддержки Service-Name. В команды `vlan rule add/rm` добавляется поддержка PPPoE и Service-Name.  
Добавление правила обработки PPPoE для указанного диапазона <Range> VLAN/QinQ:

```
vlan rule add <Range> pppoe [enable | drop | pass | delay N]
```

Добавление правила обработки PPPoE Service-Name для указанного диапазона <Range> и VLAN/QinQ:

```
vlan rule add <Range> pppoe sname <Service-Name> [enable | drop | pass | delay N]
```

Здесь <Service-Name> — имя PPPoE Service-Name в одинарных или двойных кавычках (можно без кавычек, если является идентификатором (`[a-zA-Z_][a-zA-Z_0-9]*`))

Разрешения:

1. `enable` - разрешена обработка PPPoE
2. `drop` - дропать пакеты PPPoE
3. `pass` - пропустить пакеты PPPoE насквозь без обработки
4. `delay N` - устанавливать PPPoE-сессию с задержкой в N секунд ( $0 < N < 16$ )
7. [IPFIX] Добавлена возможность отправки данных по протоколу UDP больше размера MTU (с IP фрагментацией)
8. [DNS] Добавлены параметры `ajb_save_dns_answer_types` и `ajb_save_dns_request_types`, которые позволяют задавать типы DNS запросов/ответов для записи в файл и отправку по IPFIX

9. [IPFIX] Исправлена ошибка установки таймаута отправки данных по умолчанию
10. [DHCP-Dual] Исправлено: некорректное формирование IPv6 PD-префикса для адресов из Framed-IPv6-Pool
11. [DHCP-Dual] Исправлено: падение при включении трассировки по mac bras\_dhcp\_trace\_mac
12. [DHCP-Dual] Исправлено: последовательность запроса адресов DHCPv6, затем DHCPv4 приводила к лишней авторизации
13. [DHCP-Dual] Исправлено: трассировка ответов DHCPv6, если MAC-адрес стоит на трассировке
14. [DNS] Добавлена утилита dic2dns. [Описание](#)

## Изменения в версии 14.2 БЕТА3

### DPI

1. [DPI] Добавлен разбор туннеля GRE ERSPAN для режима check\_tunnels=1
2. [DPI] Сообщение "Can't allocate record http\_state" теперь выводится раз в 50000
3. [DPI] Добавлена проверка флага MARK2 для переопределения в QUIC\_UNKNOWN\_MARKED когда протокол QUIC еще в процессе определения SNI. [Описание](#)

### BRAS

1. [BRAS][Router] Изменено вычитывание Linux route table при старте роутера. [Описание](#)

## Изменения в версии 14.2 БЕТА2.1

### NAT

1. [CG-NAT] Изменения оптимизации NAT.

## Изменения в версии 14.2 БЕТА2

### DPI

1. [DPI][DNS] Исправлена ошибка работы 19 услуги с IPv6 трафиком

### BRAS

1. [BRAS][DHCP-Dual] Добавлен учет Lease-Time  
В режиме DHCP Dual критически важно, чтобы Session-Timeout был в несколько раз (минимум в 4 раза) больше, чем Lease-Time. Если это условие нарушается, то Lease-Time полагается равным 1/4 от Session-Timeout.  
Lease-Time берется из ответа RADIUS-авторизации (в порядке уменьшения приоритета):

1. атрибут DHCP-IP-Address-Lease-Time;
2. атрибут VasExperts-DHCP-Option-Num, задающий опцию 51;
3. DHCP-опция 51, если адрес распределен из Framed-Pool.

Если Lease-time не задан ни одним из вышеперечисленных способов, то полагается равным 1/16 от Session-Timeout.

Минимальные значения:

- Session-Timeout — 600 секунд
- Lease-Time — 60 секунд

## NAT

1. [CG-NAT] Добавлена поддержка отключения кэша белых адресов для экспорта NAT трансляций. Настройка `nat_dstaddr_cache_size=0` в `/etc/dpi/fastdpi.conf`

## Изменения в версии 14.2 BETA1

### DPI

1. [DPDK] Переход на новую версию DPDK 25.11. [Описание](#)
2. [DPI][NAT] Оптимизация при переполнении кэша приватный-публичный
3. [CLI][VLAN] В команду `vlan rule dump` добавлен параметр, определяющий, какой тип правил выводить: `vlan rule dump [type]`  
`type` — rule type: perm, dhcp, all (default)  
Вывести разрешения для VLAN:

```
vlan rule dump perm
```

Вывести правила только для DHCP:

```
vlan rule dump dhcp
```

Вывести все правила:

```
vlan rule dump
```

4. [CLI][DPI] Вывод команды `fdpi_cli dump flow cache format` расширен новыми полями. [Описание](#)
5. [BALANCER] Добавлена возможность использования `vlan rule` для фильтрации пакетов
6. [DPDK] Добавлено: новая опция `dpdk_max_memzone [cold]` — Установка DPDK max memzone count. По умолчанию, в DPDK `max memzone count = 5120` (зависит от версии DPDK)  
0 — использовать default-значение, зашитое в DPDK. Устанавливать значение больше имеет смысл для huge-конфигураций со многими картами, если на старте fastDPI получаем ошибку "Number of requested memzone segments exceeds maximum 5120"
7. [CLI][DHCP-Dual] Добавлено: поддержка команды `dhcp show stat vrf`
8. [DPDK] Новый engine `dpdk_engine=7` с поддержкой явного указания диспетчеров  
Данный движок поддерживает гетерогенные конфигурации, когда в одном кластере

находятся порты разного типа — например, in-dev 100G порт, out-dev — несколько 10G портов.

Диспетчеры задаются в опциях `dppdk_dispatch`:

```
dppdk_dispatch=<список-портов>[;params]*
```

- <Список-портов> задает, какие порты обслуживает данный диспетчер
- `params` — дополнительные опции данного диспетчера. Доступные опции:
  - `rss=N` — включение RSS на всех портах данного диспетчера; будет создано N диспетчеров — по одному на каждую rx-очередь.
  - `mempool_size=N` — размер `mbuf_pool` для данного диспетчера. Каждый `dppdk_dispatch` имеет свой `mempool`, размер пулов может быть разным для разных `dppdk_dispatch`.

Опций `dppdk_dispatch` может быть много, каждая описывает отдельный диспетчер (или группу диспетчеров, если задано `rss`). Каждый порт кластера должен входить ровно в один параметр `dppdk_dispatch`. Для on-stick в `dppdk_dispatch` описывается \*базовый\* физический порт, а не on-stick порты, основанные на нем.

Фатальными ошибками конфигурации считаются следующие случаи:

- порт кластера не входит ни в какой <список-портов> параметра `dppdk_dispatch`
- порт кластера входит в <список-портов> нескольких различных параметров `dppdk_dispatch`
- в <списке-портов> параметра `dppdk_dispatch` перечислены порты из разных кластеров, — каждый диспетчер должен обслуживать порты только одного кластера

Данный движок является универсальным в том смысле, что через него могут быть выражены все остальные движки, например:

```
dppdk_engine=0: один диспетчер на все порты
  in_dev=port1:port2
  out_dev=port3:port4
  dppdk_dispatch=port1,port2,port3,port4

dppdk_engine=1: диспетчер на направление
  in_dev=port1:port2
  out_dev=port3:port4
  dppdk_dispatch=port1,port2
  dppdk_dispatch=port3,port4

dppdk_dispatch=2: диспетчер на направление с поддержкой RSS
  dppdk_rss=4
  in_dev=port1:port2
  out_dev=port3:port4
  dppdk_dispatch=port1,port2;rss=4
  dppdk_dispatch=port3,port4;rss=4

dppdk_engine=3: диспетчер на мост
  in_dev=port1:port2
  out_dev=port3:port4
```

```
dpdk_dispatch=port1,port3
dpdk_dispatch=port2,port4
```

```
dpdk_engine=4: диспетчер на порт
  in_dev=port1:port2
  out_dev=port3:port4
  dpdk_dispatch=port1
  dpdk_dispatch=port2
  dpdk_dispatch=port3
  dpdk_dispatch=port4
```

```
dpdk_engine=6: диспетчер на мост с поддержкой RSS
  dpdk_rss=4
  in_dev=port1:port2
  out_dev=port3:port4
  dpdk_dispatch=port1,port3;rss=4
  dpdk_dispatch=port2,port4;rss=4
```

Для диспетчера необходимо указывать используемый mempool.

- Описатель mempool — только для dpdk\_engine=7  
Формат:

```
dpdk_mempool=name=<name>;size=N
```

name задает имя mempool (max 15 символов)

size задает размер (число элементов) mempool

Оба параметра являются обязательными

Опций dpdk\_mempool может быть много, каждая описывает отдельный mempool

```
dpdk_mempool=...
```

- Описатель диспетчера — только для dpdk\_engine=7  
Формат:

```
dpdk_dispatch=<список-портов>;mempool=<имя>[;params]*
```

<Список-портов> задает, какие порты обслуживает данный диспетчер

params — дополнительные опции данного диспетчера. Доступные опции:

mempool=<имя> — задает имя mempool для данного диспетчера  
(обязательный параметр)

rss=N — включение RSS на всех портах данного диспетчера; будет создано N диспетчеров, по одному на каждую rx-очередь.

Опций dpdk\_dispatch может быть много, каждая описывает отдельный диспетчер (или группу диспетчеров, если задано rss)

```
dpdk_dispatch=...
```