

# Содержание

<b>Атрибуты авторизации абонента</b> .....	3
<b><i>VasExperts-Policing-Profile</i></b> .....	4
1. Предконфигурированный профиль .....	4
2. Полисинг с передачей абсолютных значений (упрощённый формат) .....	4
3. Полисинг с передачей абсолютных значений (расширенный формат HTB) .....	5
<b><i>VasExperts-Service-Profile</i></b> .....	7
Пример 1. Подключение NAT .....	7
Пример 2. Подключение белого списка .....	7
Пример 3. Переопределение классов трафика и полисинг .....	7
<b><i>VasExperts-Enable-Service</i></b> .....	8
<b><i>VasExperts-Multi-IP-User</i></b> .....	9
<b><i>VasExperts-UserName</i></b> .....	9
<b><i>VasExperts-Restrict-User</i></b> .....	9
<b><i>VasExperts-Enable-Interconnect</i></b> .....	10
<b>Оptionальные Radius-атрибуты</b> .....	10
<b><i>Session-Timeout</i></b> .....	10
<b><i>Acct-Interim-Interval</i></b> .....	10
<b><i>Idle-Timeout</i></b> .....	10
<b><i>Class</i></b> .....	11



# Атрибуты авторизации абонента

Помимо информации о том, авторизован или нет пользователь, fastPCRF необходимы следующие данные в ответ на Access-Request:

- IP-адрес абонента.
- Профиль полисинга пользователя.
- Профили пользователя по услугам fastDPI.
- Услуги, подключенные пользователю.
- Тип пользователя: количество IP-адресов, связанных с пользователем.
- Логин пользователя — атрибут User-Name или VasExperts-UserName

СКАТ поддерживает IPv4 и IPv6-адресацию абонентов. Для IPv4-абонентов в ответе должен быть атрибут Framed-IP-Address, задающий IPv4-адрес, причем этот адрес должен совпадать с тем, который был указан в Access-Request. Если значения Framed-IP-Address в запросе и ответе различаются, это считается ошибкой.



Некоторые биллинговые системы не могут отдать в Access-Accept атрибут Framed-IP-Address в режиме L3-авторизации. Для таких случаев предусмотрена fastpcrf.conf-опция radius\_framed\_ip\_from\_request=1: при таком значении в случае отсутствия в ответе атрибута Framed-IP-Address fastPCRF берет его из запроса авторизации.

Для IPv6-абонентов в Access-Accept/Reject поддерживаются атрибуты:

- Framed-IPv6-Address — задает IPv6-адрес абонента. Значение данного атрибута в ответе должно совпадать со значением в запросе.
- Framed-IPv6-Prefix — задает IPv6-префикс подсети абонента. Размер префикса и его значение в ответе должен совпадать с запросом.
- Framed-IPv6-Pool — имя пула. Необязательный атрибут. Если этот атрибут указан в ответе, то он будет передаваться во всех Accounting-Request.

Для IPv6 ответ обязательно должен содержать один из атрибутов Framed-IPv6-Address или Framed-IPv6-Prefix (или оба сразу). При этом СКАТ интерпретирует атрибут Framed-IPv6-Address как префикс подсети, не учитывая младшие биты адреса (напомним, что пока СКАТ умеет работать только с префиксами одинакового размера, задаваемым параметром [ipv6\\_subnetwork](#)).



В СКАТ IPv6-адреса **обязательно** должны быть связаны с логином абонента, поэтому в ответе должен быть один из атрибутов, задающий логин абонента.

Также поддерживаются Dual Stack абоненты: запрос Access-Request всегда идет либо по IPv4, либо по IPv6-адресу, но ответ может содержать оба адреса в атрибутах Framed-IP-Address и Framed-IPv6-Address/Framed-IPv6-Prefix. Это значит, что абоненту назначен как IPv4-адрес, так и IPv6-префикс. При этом такой абонент не считается по умолчанию multi-bind абонентом: в СКАТ с каждым абонентом, даже single-bind, может быть связана пара: IPv4-адрес

и IPv6-префикс.

Профиль полисинга и список подключенных услуг передаются в Vendor-Specific атрибуте [26] ответов.

VENDOR	VasExperts	43823	
BEGIN-VENDOR	VasExperts		
ATTRIBUTE	VasExperts-Policing-Profile	1	string
ATTRIBUTE	VasExperts-Service-Profile	2	string
ATTRIBUTE	VasExperts-Enable-Service	3	string
ATTRIBUTE	VasExperts-Multi-IP-User	4	integer
ATTRIBUTE	VasExperts-UserName	5	string
ATTRIBUTE	VasExperts-Restrict-User	7	byte или integer
END-VENDOR	VasExperts		



Необходимо создать услуги и полисинг, которые в дальнейшем будут передавать с помощью Radius атрибутов от биллинга. [Пример настройки полисинга \(тарифный план\) и Captive Portal](#), которые минимально необходимы для старта.

## VasExperts-Policing-Profile

Атрибут VasExperts-Policing-Profile передается в ответе Access-Accept и отвечает за применение правил полисинга.



В ответе Access-Accept должно присутствовать **не более одного** атрибута VasExperts-Policing-Profile.

Варианты задания профиля:

### 1. Предконфигурированный профиль

Строковый атрибут, задающий имя [профиля полисинга](#):

```
VasExperts-Policing-Profile = "50Mbps"
```

### 2. Полисинг с передачей абсолютных значений (упрощённый формат)

```
VasExperts-Policing-Profile = "BV##100000#100000#+++ -++++"
```

где:

- BV — By Value, применяется HTB алгоритм, где ceil для каждого класса будет равен root

rate

- ##100000 — лимит от Интернета к абоненту (download), кбит/с
- #100000 — лимит от абонента к Интернету (upload), кбит/с
- #+++-++++ — разрешение классов cs0 ... cs7, rate(cs3)=0 — класс cs3 заблокирован, остальные классы используют root-скорость с возможностью заимствования (HTB)

### 3. Полисинг с передачей абсолютных значений (расширенный формат HTB)

В СКАТ DPI используются [два типа полисинга](#) — HTB (Hierarchical Token Bucket) и TBF (Token Bucket Filter). В приведённых ниже примерах используется алгоритм **HTB** с ограничением минимальной (rate) и максимальной скорости по каждому классу (ceil).

Профиль полисинга определяет общую скорость (root) и скорость для каждого класса трафика (cs0 ... cs7)

Позволяет задать параметры rate и ceil для каждого класса отдельно, содержит 36 параметров.



Нулевые значения не передаются

Формат (передается одной строкой без переносов):

```
VasExperts-Policing-Profile =  
"BR##4#<RIN><ROUT><CIN><COUT><RIN0><ROUT0><CIN0><COUT0><RIN1><ROUT1><CIN1><C  
OUT1><RIN2><ROUT2><CIN2><COUT2><RIN3><ROUT3><CIN3><COUT3><RIN4><ROUT4><CIN4>  
<COUT4><RIN5><ROUT5><CIN5><COUT5><RIN6><ROUT6><CIN6><COUT6><RIN7><ROUT7><CIN  
7><COUT7>
```

где:

- BR — By Rates
- ##4# — используется HTB, передаются rate и ceil для inbound и outbound
- <RIN> — kbps root rate (#8192к), общая скорость для всех классов для трафика Inbound (download)
- <ROUT> — kbps root rate (#8192к), общая скорость для всех классов для трафика Outbound (upload)
- <RIN0> ... <RIN7> — kbps class rate, минимальная скорость по классам для трафика Inbound (download)
- <CIN0> ... <CIN7> — kbps ceil rate, максимальная скорость по классам для трафика Inbound (download)
- <ROUT0> ... <ROUT7> — kbps class rate, минимальная скорость по классам для трафика Outbound (upload)
- <COUT0> ... <COUT7> — kbps ceil rate, максимальная скорость (ceil) по классам для трафика Outbound (upload)



Сумма rate по классам должна быть  $\leq$  root rate



Ceil каждого класса должен быть  $\leq$  root rate

## Пример

Для удобства разделен на строки, но нужно передавать одной строкой.

```
VasExperts-Policing-Profile = "BR##4#  
8192k8192k65M65M  
1024k1024k8192k8192k  
1024k1024k8192k8192k  
1024k1024k8192k8192k  
1024k1024k8192k8192k  
1024k1024k8192k8192k  
1024k1024k8192k8192k  
1024k1024k8192k8192k  
1024k1024k8192k8192k"
```

## Проверка на DPI

```
sudo fdpi_ctrl list --policing --ip <ip> | sed 's/\s/\n/g'
```

Пример вывода:

```
htb_inbound_root=rate 8192kbit  
htb_inbound_class0=rate 1024kbit ceil 8192kbit  
htb_inbound_class1=rate 1024kbit ceil 8192kbit  
htb_inbound_class2=rate 1024kbit ceil 8192kbit  
htb_inbound_class3=rate 1024kbit ceil 8192kbit  
htb_inbound_class4=rate 1024kbit ceil 8192kbit  
htb_inbound_class5=rate 1024kbit ceil 8192kbit  
htb_inbound_class6=rate 1024kbit ceil 8192kbit  
htb_inbound_class7=rate 1024kbit ceil 8192kbit  
  
htb_root=rate 8192kbit  
htb_class0=rate 1024kbit ceil 8192kbit  
htb_class1=rate 1024kbit ceil 8192kbit  
htb_class2=rate 1024kbit ceil 8192kbit  
htb_class3=rate 1024kbit ceil 8192kbit  
htb_class4=rate 1024kbit ceil 8192kbit  
htb_class5=rate 1024kbit ceil 8192kbit  
htb_class6=rate 1024kbit ceil 8192kbit  
htb_class7=rate 1024kbit ceil 8192kbit
```

# VasExperts-Service-Profile

Строковый параметр, задающий имя профиля для конкретной услуги fastDPI.  
Формат строки:

```
service_id:profile_name
```

Где:

1. service\_id — число, [идентификатор услуги fastDPI](#);
2. profile\_name — строка, [имя профиля по услуге](#).



Ответ на авторизацию может содержать ноль или более атрибутов VasExperts-Service-Profile — по одному атрибуту для каждой услуги.

## Пример 1. Подключение NAT

Подключение NAT (11 услуга) с профилем "cgnat".

```
VasExperts-Service-Profile="11:cgnat"
```

## Пример 2. Подключение белого списка

Подключение [Белого списка](#) и [Captive Portal, HTTP redirect](#) (16 услуга) с указанием параметров страницы для редиректа "<http://info.com>" и разрешенного списка IP адресов `"/var/lib/dpi/ip_list.bin"`.

```
VasExperts-Service-Profile =  
"16:BV##/var/lib/dpi/ip_list.bin#http://info.com"
```

## Пример 3. Переопределение классов трафика и полисинг

[Услуга 18 \(Полисинг по сессии\)](#) используется для переопределения классов трафика (cs0 ... cs7) и задания сессионного полисинга для конкретного абонента.

Профиль определяет распределение протоколов DPI по классам трафика.  
Классы назначаются индивидуально для каждого абонента.

Формат:

```
VasExperts-Service-Profile =  
"18:BP##profile_name#/share/#IMSI_number.dscp#IMSI_number.tbf"
```

где:

- 18 — переопределение классов трафика и полисинг по сессии
- BP — By Parameters
- ##profile\_name — имя профиля, создаётся динамически, отображается в GUI и CLI
- #/share/ — директория на fastPCEF. Доступна по сети, передается только имя файла, а сам файл считывается из оригинальной директории. Содержит исходные (несконвертированные в бинарные) файлы .txt для удобства проверки
- #IMSI\_number.dscp — файл соответствия протоколов и классов трафика. Определяет, в какой класс (cs0 ... cs7) помещается трафик. Поддерживает флаг **drop** для блокировки
- #IMSI\_number.tbf — файл с параметрами сессионного полисинга. Задаёт правила полисинга для отдельных протоколов. Используется для ограничения скорости на уровне сессии

Пример:

```
VasExperts-Service-Profile
=18:BP##250019500475292#/share/#250019500475292.dscp#250019500475292.tbf
```

## VasExperts-Enable-Service

Строковый параметр, задающий включение/отключение конкретной услуги, для которой не требуется профиль. Формат строки:

```
service_id:flag
```

где:

- service\_id — число, идентификатор услуги fastDPI
- flag — признак включения/отключения услуги. Допустимые значения:
  - 1, on, enabled — услуга включена
  - 0, off, disabled — услуга отключена

Пример подключенной услуги: 5: on

Пример отключенной услуги: 5: off

В CoA для отключения услуги следует использовать данный атрибут. Например, отключение услуги 5 выглядит так: `VasExperts-Enable-Service="5:off"`. Тогда как включение услуги 5 с профилем `my_white_list` — так: `VasExperts-Service-Profile="5:my_white_list"`

**!** Важное замечание: для услуг действует правило “что явно не подключается в ответе, то будет отключено”, то есть если услуга явно не помечена как `enabled` (или не задан профиль услуги), то услуга считается отключенной. Но для услуги 4 (**black list**, фильтрация запрещенного трафика) действует более строгое правило: эту услугу нужно явно отключать для пользователя, если требуется, то есть для отключения услуги 4 “black-list” в ответе Radius-сервера должен явно присутствовать атрибут `VasExperts-Enable-Service="4:off"`.

По умолчанию подключение услуги 4 “black-list” регулируется [глобальными настройками fastDPI](#). Услуга 4 обычно глобально включена, чтобы не нарушать федеральное законодательство.

## VasExperts-Multi-IP-User

Признак, связано ли с данным абонентом множество IP-адресов или только один. Данный атрибут может быть либо байтом, либо 32-битовым числом. Значение 1 говорит о том, что данному абоненту может быть сопоставлено несколько IP-адресов (корпоративный клиент), значение 0 – только один IP-адрес. Если в PDU отсутствует атрибут VasExperts-Multi-IP-User, считается, что абоненту сопоставлен только один IP-адрес.

Если абоненту предоставлен признак VasExperts-Multi-IP-User=1, то свойства (подключенные услуги и полисинг) применяются ко всем IP-адресам абонента, ключом выступает логин абонента. Следует учитывать, что СКАТ авторизует **каждый** IP-адрес абонента: например, если с абонентом связано 10 IP-адресов, то для **каждого** адреса будет послан запрос авторизации Access-Request. Ожидается, что ответ для каждого IP-адреса multi-IP-абонента будет содержать один и тот же набор подключенных услуг и одни и те же профили. Ответ на авторизацию каждого из 10 IP-адресов будет применен к **логину** абонента, то есть все IP-адреса данного логина получают один и тот же набор услуг и один и тот же полисинг.

## VasExperts-UserName

Имя (логин) абонента.

Этот атрибут введен для некоторых биллинговых систем, которые физически не могут отдать в ответе Access-Accept/Reject атрибут User-Name, но могут отдать любой VSA-атрибут.



Если в ответе присутствуют оба атрибута - User-Name и VasExperts-UserName, — то предпочтение отдается значению VasExperts-UserName.

## VasExperts-Restrict-User

Признак блокировки абонента.

Для СКАТ ответ Access-Accept означает, что абонент не заблокирован, а для заблокированных абонентов должно возвращаться Access-Reject со специальными атрибутами абонента, задающими ограничения. Но некоторые реализации радиус-клиентов не имеют возможности вернуть в Access-Reject атрибуты абонента. Для таких реализаций предусмотрен атрибут VasExperts-Restrict-User:

- значение 0 - абонент не заблокирован
- значение 1 - абонент заблокирован

В случае VasExperts-Restrict-User=1 СКАТ обрабатывает Access-Accept как Accept-Reject.

## VasExperts-Enable-Interconnect

Признак разрешения/запрета замыкания локального трафика для абонента.

Данный атрибут относится к [L2 BRAS](#). Если включен режим замыкания локального трафика (local interconnect) `bras_terminate_local=1`, L2 BRAS замыкает любых двух локальных абонентов. С помощью данного атрибута можно запретить интерконнект: `VasExperts-Enable-Interconnect=0`. Пакет от одного локального абонента к другому будет дропнут, если хотя бы для одного из абонентов интерконнект запрещен.

По умолчанию `VasExperts-Enable-Interconnect=1` и если в `fastdpi.conf` задано `bras_terminate_local=1`, то интерконнект разрешен. Данным атрибутом можно запретить интерконнект для конкретного абонента, указав `VasExperts-Enable-Interconnect=0`.

## Опциональные Radius-атрибуты

Помимо перечисленных выше VSA, СКАТ поддерживает в Access-Accept/Access-Reject следующие стандартные Radius-атрибуты. Все они являются опциональными.

### Session-Timeout

Задаёт продолжительность действия авторизации абонента в секундах. По истечении данного времени СКАТ пошлёт повторный запрос Access-Request авторизации.

По умолчанию длительность сессии задается в `fastdpi.conf`, параметр `auth_expired_timeout` (в **минутах**).

### Acct-Interim-Interval

Задаёт интервал времени обновления Accounting-статистики в секундах для данного абонента.

Interim interval по умолчанию задается в `fastpcrf.conf`, параметр `radius_acct_interim_interval`.



Явное задание `Acct-Interim-Interval = 0` в ответе RADIUS отключает отправку Interim-Update.

### Idle-Timeout

Задаёт интервал неизменности данных аккаунтинга, по истечении которого accounting-сессия

считается закрытой из-за неактивности абонента. Idle timeout по умолчанию задается fastpcrf.conf-параметром radius\_acct\_idle\_timeout. Как определять неактивность абонента, задается fastpcrf.conf-параметром acct\_check\_idle\_mode. Описание параметров находится в разделе [Настройки fastPCRF](#).

## Class

При наличии этого атрибута, он будет копироваться во все Accounting-Request PDU. СКАТ не анализирует значение этого атрибута.

[Особенности Access-Reject](#)