

Содержание

7 QoE Stor Module	3
Introduction	3
Architecture	3
Installation and Update	3
Recommendations on device to be used for installing QoE Stor Module	3
Version Information	5
Installation	6
Upgrading	6
Configuration	7
Configuring the dictionaries	14
asnum_local_dic and subnets_local_dic dictionaries	15
subscribers_dic, switches_dic, crc_dic dictionaries	15
Справочники urlcats_dic и urlcats_host_dic	17
Проблемы и решения	17
Не работает, хотя все установили по инструкции	17
Выполнили yum -y update, не запускаются ресиверы	19
SQL и выгрузка данных в CSV, JSON, TabSeparated	19

7 QoE Stor Module

QoE analytics data collection and storage module

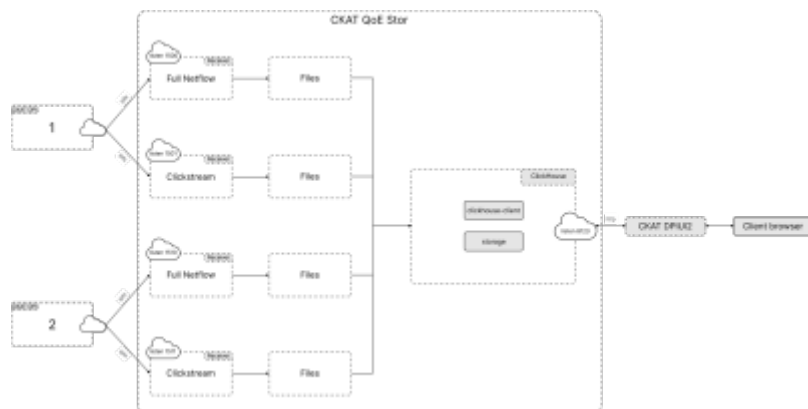
Introduction

The module is designed to collect and store Neflow and Clickstream data. Data is used to analyze QoE in DPIUI2.

Architecture

Data from the VAS Experts DPI is received on several sockets (tcp or udp) using [utility designed to collect IPFIX stream data](#).

The collected data is stored in the ClickHouse database.



Installation and Update

Recommendations on device to be used for installing QoE Stor Module



Do not install the module on the same server with the DPI platform!

Minimum requirements

For the subsystem, you can use hardware or virtual machines with the characteristics listed below:

1. Processor (CPU) 2.5 ГГц - 1 pce
2. RAM - at least 16 GB
3. Hard disk drive (SSD is highly desirable) - at least 500 GB

4. Operating system - Cent OS 6.4 - 7.x+
5. Network interface (NIC) - at least 1 Gbps

10 Gbps average daily traffic generates approximately 25 GB of data per hour in the QoE Store

Processor

SSE 4.2 instruction set support is required.

Choose processors with a large number of cores. Clock frequency is less important. For example, 16 cores with 2,600 MHz are better than 8 cores at 3,600 MHz.



Do not disable Hyper-threading and Turbo-Boost.

RAM

RAM should be no less than the amount of data requested.

The more RAM server has, the better performance will be achieved when building reports.

The more memory, the less disk system is stressing.

Minimum prerequisites is 16 GB.

Always disable the swap file.

Disk

Required disk space is at least 16 GB for every storage day, it's actual value depends on daily traffic.

If your budget makes you possible to use SSD, use it. Otherwise use HDD. SATA HDDs 7200 RPM will be suitable.

When using HDD, you can combine them into RAID-10, RAID-5, RAID-6 or RAID-50.

Most of the data is stored in the `/var/lib/clickhouse/` directory. You can mount a drive/partition on this directory.

Temporary data (ipfix dumps) is stored in the `/var/qoestor/backend/dump/` directory. 50 GB be sufficient for this.

Maintenance advice from Yandex ClickHouse

You could familiarize yourself with the contents of maintenance advice from Yandex ClickHouse by

following the <https://clickhouse.yandex/docs/ru/operations/tips/> link.

Version Information

Version v.1.0.9 (21.02.2019)

- Bug concerning incorrect recognition of trunk switches is fixed
- protocols_dic dictionary is updated

Version v.1.0.7 hot fixes (24.12.2018)

- ipfix re-export feature is added and is available through the ipfixreceiver configuration file: IPFIX_FULLFLOW_EXPORT and IPFIX_CLICKSTREAM_EXPORT

Version v.1.0.6 hot fixes (04.12.2018)

- Bugs concerning the work of the subnets_local_dic dictionary (such as "A call to function range would produce 12884901882 array elements")
- ipfixreceiver2 receiver configuration has been fixed (FileWriter queue is full. Records dropped.)

Version v.1.0.5 (03.12.2018)

- Host Category Dictionaries
- Transition to the ipfixreceiver2

1. Do not forget to update the installation script (in the Installation section) before upgrading. To upgrade, use the installation script.
2. After the upgrade has been finished, check whether the receivers are running:

```
netstat -nlp | grep 1500
```



and

```
netstat -nlp | grep 1501
```

ipfixreceiver2 should listen corresponding sockets.

3. If receivers are not running, execute the /var/qoestor/backend/qoestor-config.sh script

Version v.1.0.4 (02.11.2018)

- Preaggregation is implemented resulting in reducing the netflow by 6-7 times and clickstream by 3 times
- The following dictionaries are implemented: subscribers, switches, autonomous systems (AS), crc
- The following features are added: identification the traffic direction and subscribers filtering (separation of IP hosts and IP subscribers) by AS and CIDR. This option is meaningful only if the VAS Experts DPI is implemented using the mirror connection scheme.

This QoE Stor version works with the version of DPIUI2-2.1.5 and higher



If you have already installed version 1.0.0, then you should to delete the database due to complete version incompatibility before installing the new version. To do so issue the command:

```
clickhouse-client --query="drop database qoestor"
```

Version v.1.0.0 (20.09.2018)

- A new QoE Stor module is implemented

Installation



Before installing or upgrading, check your Internet connection. Make shure you run scripts under the root or using sudo.

For automatically installation or upgrading follow these steps:

1. Execute [clickhouse-repos_config.sh.gz](#) script. The repository will be prepared for the subsequent installation of the clickhouse database as a result.
2. Execute [fastor-rpm_install.sh.gz](#) script. It will cause the installation of the following packages: ipfixreceiver, clickhouse, fastor. All of them will be automatically configured according to the defaults.
3. Check whether the qoestor database in clickhouse is available. To do this, issue the command

```
clickhouse-client --query="show databases" | grep qoestor
```

4. If there is no database (probably the database server is not running), you should to create it by issuing the following command

```
clickhouse-client -n < /var/qoestor/backend/etc/db/qoestor.sql
```

Upgrading

Upgrading is performed using the same scripts as in the [installation](#) section.

If receivers stopped after you have executed the

```
yum -y update
```

command, please refer to the troubleshooting section following the [link](#).

Configuration

ipfix receivers configuration

ipfix receivers configuration is implemented through the .env file:

```
/var/questor/backend/.env
```

Default configuration looks like:

```
#Ipfix form DPI 0
IPFIX_FULLFLOW_PORT_TYPE[0]=tcp
IPFIX_FULLFLOW_PORT[0]=1500
#IPFIX_FULLFLOW_EXPORT[0]=10.0.0.2/9920/tcp,10.0.0.3/3440/udp

IPFIX_CLICKSTREAM_PORT_TYPE[0]=tcp
IPFIX_CLICKSTREAM_PORT[0]=1501
#IPFIX_CLICKSTREAM_EXPORT[0]=10.0.0.2/9921/tcp,10.0.0.3/3441/udp

#Traffic direction definition
# 0 - as is
# 1 - by AS (for fullflow only)
# 2 - by CIDR (for fullflow and clickstream)
# 3 - by both: AS and CIDR
# 4 - any: AS or CIDR
TRAFFIC_DIR_DEF_MODE=0

#Subscriber filter
# 0 - no filter
# 1 - by AS (for fullflow only)
# 2 - by CIDR (for fullflow and clickstream)
# 3 - by both: AS and CIDR
# 4 - any: AS or CIDR
SUBSCRIBER_FILTER_MODE=0
```

In the configuration above fullflow and clickstream receivers are listening 1500 and 1501 socket respectively. «0» in array subscript means that the receiver get the data from DPI number 0.



It is better to prefer tcp over udp because udp packets can be lost when the MTU is exceeded.

The values `TRAFFIC_DIR_DEF_MODE = 0` and `SUBSCRIBER_FILTER_MODE = 0` mean that there is no need to calculate the traffic direction and apply filters to subscribers.

The `IPFIX_FULLFLOW_EXPORT` and `IPFIX_CLICKSTREAM_EXPORT` parameters allow you to configure export to third-party receivers. Format for use: `ip/port/proto[,ip/port/proto]`.



If the configuration has changed, you should run the `/var/qoestor/backend/qoestor-config.sh` script

The following example shows how to configure data reception from several DPIs

```
#Ipfix form DPI 0
IPFIX_FULLFLOW_PORT_TYPE[0]=tcp
IPFIX_FULLFLOW_PORT[0]=1500

IPFIX_CLICKSTREAM_PORT_TYPE[0]=tcp
IPFIX_CLICKSTREAM_PORT[0]=1501

#Ipfix form DPI 1
IPFIX_FULLFLOW_PORT_TYPE[1]=tcp
IPFIX_FULLFLOW_PORT[1]=1510

IPFIX_CLICKSTREAM_PORT_TYPE[1]=tcp
IPFIX_CLICKSTREAM_PORT[1]=1511

#Ipfix form DPI 2
IPFIX_FULLFLOW_PORT_TYPE[2]=tcp
IPFIX_FULLFLOW_PORT[2]=1520

IPFIX_CLICKSTREAM_PORT_TYPE[2]=tcp
IPFIX_CLICKSTREAM_PORT[2]=1521
```

The following example corresponds to the situation when you need to identify subscribers by CIDR

This configuration makes sense only when the VAS Experts DPI is installed using port mirroring.

```
TRAFFIC_DIR_DEF_MODE=2
SUBSCRIBER_FILTER_MODE=2
```

Be sure to configure `subnets_local_dic` dictionary for this configuration example!

The following example corresponds to the situation when export to third-party receivers is configured

```
IPFIX_FULLFLOW_PORT_TYPE[0]=tcp
IPFIX_FULLFLOW_PORT[0]=1500
IPFIX_FULLFLOW_EXPORT[0]=10.0.0.2/1600/tcp
```

```
IPFIX_CLICKSTREAM_PORT_TYPE[0]=tcp
IPFIX_CLICKSTREAM_PORT[0]=1501
IPFIX_CLICKSTREAM_EXPORT[0]=10.0.0.2/1601/tcp
```

Restarting the receivers

All receivers can be restarted using the command:

```
/var/qeстор/backend/qeстор-config.sh
```

If you need to restart the receivers one by one, you can do this by restarting corresponding systemd service units, for example

- For CentOS 7

```
systemctl restart qeстор_fullflow_0.service
systemctl restart qeстор_clickstream_0.service
```

- For CentOS 6

```
service qeстор_fullflow_0 stop
service qeстор_clickstream_0 stop
service qeстор_fullflow_0 start
service qeстор_clickstream_0 start
```

Stopping the receivers

- For CentOS 7

```
systemctl stop qeстор_fullflow_0.service
systemctl stop qeстор_clickstream_0.service
```

- For CentOS 6

```
service qeстор_clickstream_0 stop
service qeстор_fullflow_0 stop
```

Clickhouse DB stop and start

- Stop

```
sudo /etc/init.d/clickhouse-server stop
```

- Start

```
sudo /etc/init.d/clickhouse-server restart
```

DPI configuration

Export configuration

The DPI version must be at least 8.1.

You can configure ipfix export by editing the fastdpi.conf configuration file on your DPI device.

```
netflow=8
netflow_dev=em1
netflow_timeout=10
netflow_as_direction=3
netflow_full_collector_type=2
netflow_full_port_swap=0
netflow_full_collector=YOUR_QOESTOR_IP:1500
netflow_passive_timeout=20
netflow_active_timeout=60
netflow_rate_limit=120
ipfix_dev=em1
ipfix_tcp_collectors=YOUR_QOESTOR_IP:1501
```



fastdpi restart is needed for the changes to take effect:

service fastdpi restart

This can be achieved also using [DPIUI2](#). The dpiui2 version must be at least 2.1.0.

To perform configuration using DPIUI2 you should open the section DPI CONTROL → CONFIGURATION. Open the tab **Collection and analysis of statistics on protocols and directions**.

Set netflow option to **Export of complete statistics for sessions**. See figure below.



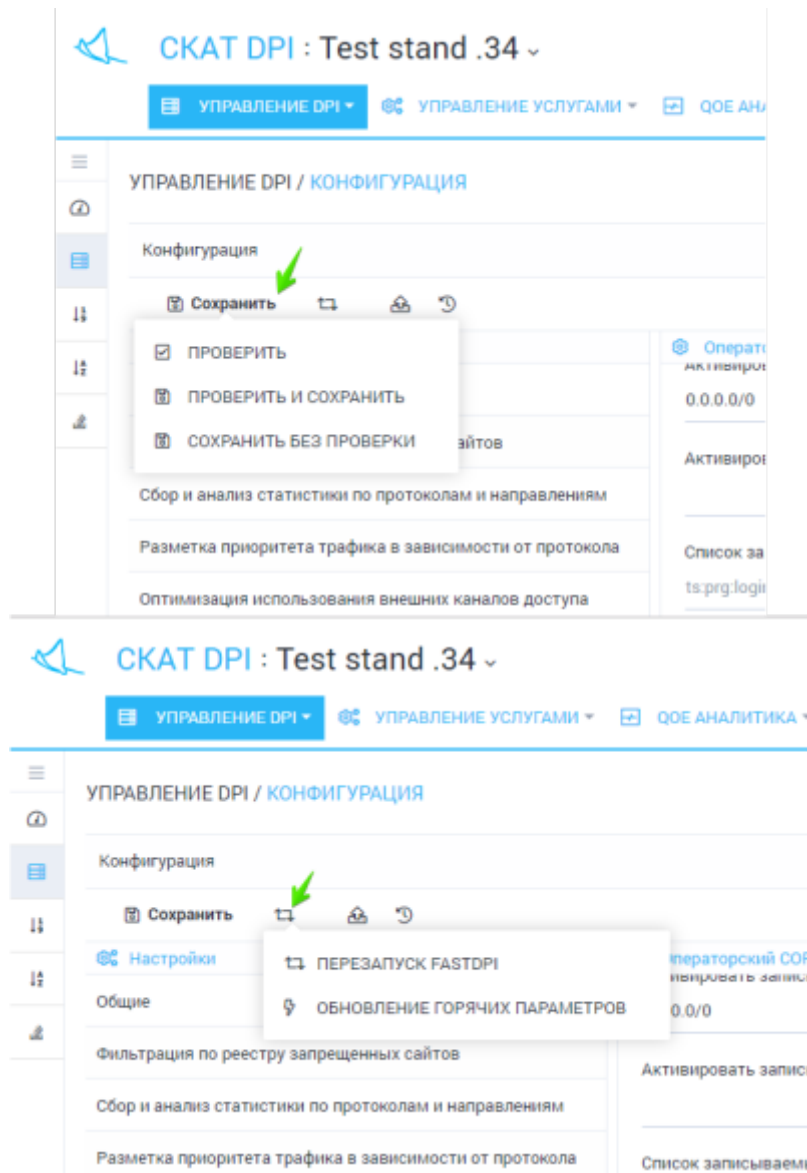
Then specify socket for fullflow receiver within "netflow_full_collector" parameter: "IP address of the netflow collector with full statistics (netflow_full_collector)". "netflow_full_collector_type" should be set to "Export ipfix to udp header", whereas "netflow_full_port_swap" should be empty or equals to "Keep original port numbers". See the figure below.



Type in the clickstream receiver socket in "ipfix_udp_collectors" parameter under the "DPI CONTROL → CONFIGURATION → Operator's SORM → IP or the domain name (: port) of the ipfix collector clickstream (ipfix_udp_collectors)". See the figure below.

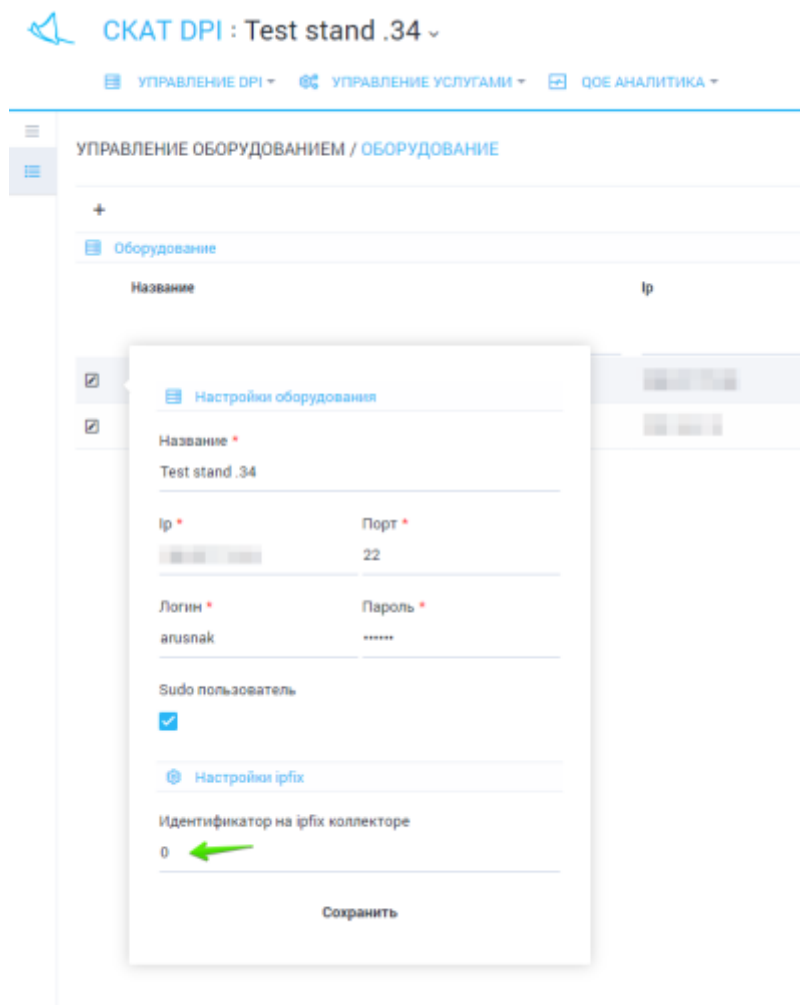
[illegible]

Press Save and then restart fast dpi. See the figure below.



ID assignment to ipfix collector

Open the section Hardware "Management → Equipment → Hardware settings". Type in an identifier within the "Id on ipfix collector" section (under the "Ipfix settings") for the ipfix collector. See the figure below.



Setting up DPIUI2 connection to the QoE Stor module

In order to browse QoE reports, you should configure the DPIUI2 connection to the QoE Stor. See the [Setting up a connection to the QoE Stor](#) section.

Configuring the dictionaries

All dictionaries are in the `/var/qoestor/backend/etc/db/` directory and have `.txt` extension

Each dictionary has its own `sample.txt` which can be used as a pattern.

All columns within the dictionaries are separated by a tab character (`\t`). The number of `\t` should be one less than the number of columns in dictionary. Please, pay close attention to this circumstance.

When files are changed, the data is loaded into the database automatically.

Some useful commands when working with dictionaries:

- Speed up data updates in directories

```
clickhouse-client --database=qoestor --query="system reload dictionaries"
```

- Check dictionaries for errors

```
clickhouse-client --database=qoestor --query="select * from
system.dictionaries"
```

- Check if there is data in the directory, for example for the subnets_local_dic

```
clickhouse-client --database=qoestor --query="select * from
subnets_local_dic"
```

asnum_local_dic and subnets_local_dic dictionaries

The lists of your local AS and local subnets are specified in these dictionaries. Dictionaries are used to identify the traffic direction (true in case the DPI is installed using mirroring) and to filter subscribers (so that the hosts IP addresses do not appear in the subscriber reports)

Example of **asnum_local_dic** dictionary

```
12345    LOCAL
65535    UNKNOWN
```

The first column is AS number, the second one is it's name (it is displayed in reports).

Example of **subnets_local_dic** dictionary

```
192.168.1.0/24    LOCAL
10.64.66.0/24     LOCAL
172.16.0.0        LOCAL
2a02:2168:aaa:bbb::2    LOCAL
```

The first column is IP address or CIDR, the second one is the name (it is NOT displayed in reports, but it is required by format).



Do not add too large subnet. Break into small ones. Highest value (limit) is 100000000

subscribers_dic, switches_dic, crc_dic dictionaries

subscribers_dic

Dictionary of subscribers.

Dictionary example

```
10.64.66.100    login    5    port1    unit_vendor    cabel    contract
services        mac
10.64.66.101    login    2    port1    unit_vendor    cabel    contract
```

services	mac						
10.64.66.102	login	3	port1	unit_vendor	cabel	contract	
services	mac						
10.64.66.103	login	4	port1	unit_vendor	cabel	contract	
services	mac						
10.64.66.104	login	5	port1	unit_vendor	cabel	contract	
services	mac						
10.64.66.105	login	5	port2	unit_vendor	cabel	contract	
services	mac						
10.64.66.106	login	5	port3	unit_vendor	cabel	contract	
services	mac						

Columns:

1. IP address
2. Login
3. Switch ID (access switch)
4. Switch port
5. Subscriber device vendor
6. Cable
7. Contract
8. Services
9. Subscriber device MAC address (is reserved for future purposes)

switches_dic

Hierarchical dictionary of devices (access switches and trunk switches)

Dictionary example

1	Switch_1	Ethernet	Region_1	Address_1	10.140.1.18	ISP_1
0	0					
2	Switch_2	Ethernet	Region_2	Address_2	10.140.2.18	ISP_1
0	0					
3	Switch_3	Ethernet	Region_3	Address_3	10.140.3.18	ISP_1
0	1	port1				
4	Switch_4	Ethernet	Region_4	Address_4	10.140.4.18	ISP_1
0	3	port1				
5	Switch_5	Ethernet	Region_5	Address_5	10.140.5.18	ISP_1
0	4	port1				

Columns:

1. Device ID UInt64
2. Device name
3. Device type
4. Region
5. Address
6. Switch IP address
7. Internet service provider

8. Indicator: trunk switch indicator (1 if so). Is not currently used, you can set 0 everywhere
9. Upstream Switch ID UInt64
10. Upstream Switch port
11. The owner

crc_dic

Справочник ошибок (CRC) на портах коммутаторов

Пример справочника

2	port1	450
5	port1	550
5	port2	500
4	port1	780

Столбцы

1. Идентификатор коммутатора
2. Порт коммутатора
3. Значение CRC

Справочники urlcats_dic и urlcats_host_dic

Справочники Категорий хостов. Предназначены для определения принадлежности хоста определённой категории.

Справочники подкачиваются автоматически с ресурсов vasexperts.ru.

Для ускорения начальной загрузки выполните

1.

```
sh /var/qoestor/backend/etc/cron_daily.sh
```
2.

```
clickhouse-client --database=qoestor --query="system reload dictionaries"
```

Проблемы и решения

Не работает, хотя все установили по инструкции

Если вы все установили и настроили по инструкции, а в разделе DPIUI2 “QoE Аналитика” пусто, то вот перечень шагов, которые стоит выполнить, прежде чем обращаться в тех. поддержку.

1. В GUI попробовать изменить период (из-за таймзон может не показывать)
2. На сервере с QoE Stor проверить, создана ли БД

```
clickhouse-client --query="show databases" | grep qoestor
```

Если БД не создана, создать ее командой

```
clickhouse-client -n < /var/qoestor/backend/etc/db/qoestor.sql
```

3. На сервере с QoE Stor проверить, есть ли данные в БД

```
clickhouse-client --query="select count(), min(flow_start_time),  
max(flow_start_time) from qoestor.fullflow"
```

и

```
clickhouse-client --query="select count(), min(time), max(time) from  
qoestor.clickstream"
```

4. На сервере с QoE Stor проверить, есть ли что-либо в дампах ресиверов

```
/var/qoestor/backend/dump/fullflow
```

и

```
/var/qoestor/backend/dump/clickstream
```

5. На сервере с QoE Stor проверить логи ресиверов в папке

```
/var/qoestor/backend/logs
```

Есть ли там что-то типа "oops! "? Обращайтесь в тех. поддержку, т.к скорее всего установлены компоненты не из нужных репо. Если там "Illegal IPFIX Message Version 0x0005", то еще раз проверьте настройки экспорта на dpi: netflow_full_collector_type установлен неверно.

6. На сервере с QoE Stor проверить, прослушиваются ли порты 1500 и 1501 командой

```
netstat -nlp | grep 1500 и netstat -nlp | grep 1501
```

Перезапустить все ресиверы на всякий случай командой

```
/var/qoestor/backend/qoestor-config.sh
```

7. Еще раз проверить [настройки экспорта ipfix на dpi](#)
8. На сервере с DPIUI2 проверить [настройки подключения GUI к QoE Stor](#)
9. На сервере с QoE Stor проверить, запущена ли СУБД ClickHouse командой

```
ps aux | grep clickhouse
```

Убедитесь, что достаточно оперативной памяти на сервере.

10. На сервере с QoE Stor проверить логи clickhouse в папке /var/log/clickhouse-server/

Если есть необходимость очистить все данные в БД, то на сервере с QoE Stor надо

1. Удалить БД командой

```
clickhouse-client --query="drop database qoestor"
```

2. Пересоздать БД командой

```
clickhouse-client -n < /var/qoestor/backend/etc/db/qoestor.sql
```

Выполнили `yum -y update`, не запускаются ресиверы

При выполнении **yum -y update** ломаются некоторые библиотеки. Ресиверы перестают запускаться.

1. Удалите `fastor` и зависимости

```
yum remove fastor ipfixreceiver libfixbuf netsa_silk netsa-python
```

2. Установите заново, используя скрипт [fastor-rpm_install.sh.gz](#)

SQL и выгрузка данных в CSV, JSON, TabSeparated

При необходимости вы можете самостоятельно без дополнительных инструментов сформировать собственные отчеты и выгрузить данные в любом формате CSV, JSON, TabSeparated.

Данные хранятся в 4 основных логах

- `qoestor.fullflow` – полный netflow лог, период хранения – 24 часа
- `qoestor.clickstream` – полный clickstream лог, период хранения – 24 часа
- `qoestor.fullflow_agg` – агрегированный netflow лог, период хранения не ограничен
- `qoestor.clickstream_agg` – агрегированный clickstream лог, период хранения не ограничен

Формат команды следующий

```
clickhouse-client --database=qoestor --query="Ваш sql тут"
```

По умолчанию данные выгружаются в формате TabSeparated.

Пример. Клиент попросил лог соединений с определенным хостом в формате CSV

```
clickhouse-client --database=qoestor --query="select * from fullflow  
prewhere flow_start_date = '2018-10-04' where (source_ipv4 = '10.64.66.100'  
or destination_ipv4 = '10.64.66.100') and host = 'google.com' ORDER BY  
flow_start_time limit 10 format CSV"
```

Подробную информацию по SQL ClickHouse смотрите по ссылке

https://clickhouse.yandex/docs/ru/query_language/select/