Содержание

Configuring Full NetFlow Export in IPFIX Format

General Required NetFlow Configuration Settings

Enabling statistics collection and export:

```
netflow=1
```

- 0 or not specified - option disabled
- 1 - export statistics by protocol (port numbers); see the section Set up NetFlow export by protocols, directions and billing for details
- 2 - export statistics by destination (autonomous system numbers); see the section Set up NetFlow export by protocols, directions and billing for details
- 4 - Export statistics for billing; see the section Set up NetFlow export by protocols, directions and billing for details
- 8 - Export full session statistics, Full NetFlow in NetFlow v5 or IPFIX format

Simultaneous export of Full NetFlow and billing statistics is enabled via the bitmask netflow=12 (8 + 4). Billing statistics are converted to RADIUS Accounting via FastPCRF when enable_acct=1 is enabled. RADIUS Accounting Configuration

You must assign a separate collector for each type to prevent data from getting mixed up!

The name of the network interface through which NetFlow statistics will be sent:

```
netflow_dev=eth2
```

IPFIX/Netflow settings can be changed without restarting fastDPI.
The `ipfix_reserved` configuration parameter allows you to reserve the necessary memory to enable or modify IPFIX/Netflow settings.
If IPFIX/Netflow settings are specified in the configuration file, memory reservation for IPFIX/Netflow is automatically enabled, and IPFIX/Netflow settings and new IPFIX/Netflow exporter types can be changed without rebooting fastDPI.

For receiving, processing, and storing IPFIX, we recommend using the QoE Store statistics collection software and the DPIUI2 graphical interface.

Any universal IPFIX collector that supports templates, or the IPFIX Receiver utility, is suitable for collecting information in the IPFIX format.

Configuration Example

See the QoE Stor: DPI Configuration section for a configuration example

General Additional NetFlow Configuration Settings

Data export interval (in seconds):

```
netflow_timeout=10
```

The default value is 30 seconds.

Session timeout:

- `netflow_passive_timeout` — the timeout period (in seconds) for session activity; after this period, if there has been no activity, the session is considered terminated and data is transmitted via it. The default value is 30 seconds.
- `netflow_active_timeout` — the time (in seconds) after which information is reported for long sessions (i.e., long sessions are effectively broken down into segments of this duration). The default value is 300 seconds.

To smooth out spikes and distribute the load more evenly across the collector, set the configuration parameter

```
netflow_rate_limit=900
```

, where 900 is the maximum NetFlow rate in Mbps.
The default value for this parameter is 0 (unlimited).

The parameter value should be set based on the following calculation: each DPI generates an IPFIX stream at a rate of 0.5% to 1% of the actual traffic rate.
Setting a value that is too low will result in data being discarded at the DPI side.
Information about this event will be recorded in the log file **/var/log/dpi/fastdpi_alert.log**.

Sending a template via IPFIX

1. TCP transport protocol.
   The template is sent once after a TCP session is established.
2. UDP transport protocol.
   By default, the template is sent every 20 seconds. This can be adjusted using the `ipfix_udp_template_timer` parameter.

Full NetFlow Configuration

Specify the IP address and port number of the **Full NetFlow** collector. You must assign a separate collector to each FastDPI to ensure that the data is not mixed with other statistics:

```
netflow_full_collector=192.168.0.1:9996
```

Specify the **Full NetFlow** export format:

```
netflow_full_collector_type=2
```

Possible values:

- **0** - Export in NetFlow5 format (default).
- **1** - Export IPFIX to a UDP collector.
- **2** - Export IPFIX to a TCP collector.

We recommend using Full NetFlow transmission in IPFIX format over TCP (parameter value 2).
The NetFlow protocol does not guarantee packet delivery (since it operates over UDP), and if the collector cannot handle the incoming data, some packets will simply be lost. Transmitting **Full NetFlow** for 10 Gbps of DPI traffic requires the collector to be capable of receiving data at a rate of at least 60 Mbps.
Check your collector's capabilities before sending **Full NetFlow** statistics to it. At the same time, when transmitting **Full NetFlow** from DPI, short-term peaks of up to 100 Mbps may occur during spikes in the number of sessions.

When sending IPFIX over UDP, errors may occur due to incorrect settings on the receiving end, which can result in data loss. Examples of such errors:

```
[ERROR      ][2026/03/12-11:52:53:559204][0x7fdeba84b400]
IPFIX_ClickStream : udp:10.16.20.183:1502 : Error socket send to
collector, rc=-1, errno=113 : No route to host
[ERROR      ][2026/03/12-11:52:53:559243][0x7fdeba84b400]
IPFIX_ClickStream : udp:10.16.20.183:1502 : Error socket send to
collector ( repeat error 2 ), now ok.
```

The `netflow_plc_stat` parameter defines the set of statistics on dropped packets to be transmitted in accordance with policing or drop rules. The parameter is a bit mask.
By default, the mask has the value ''**0x07**''—statistics on dropped data for session, subscriber, and virtual channel policing are transmitted.
Affects the calculation of the DROPPED_BYTES and DROPPED_PACKETS counters.
Values used to construct the mask:

- **0xff** - any drop is counted
- **0** - do not count
- **1** - count for session-based policing
- **2** - count for subscriber-based policing
- **4** - count for virtual channel policing

- **8** - count when packets are dropped by the protocol
- **16** - count in all other cases

The `ipfix_mtu_limit` parameter specifies the maximum size of a UDP packet when sending IPFIX. By default, it is set to the minimum MTU size of the interfaces used for transmission.

The parameter `tethering_ttl_allowed = 128:64` specifies a list of valid TTL values for traffic from the subscriber that is not considered tethering. Values are separated by a colon ':'. The number of values can be up to 256 (0–255).

Export Template in IPFIX Format (Netflow v10) for IPv4 Protocol

Export Template for IPv4						
№	**Bytes**	**Data Type**	**IANA**	**Description**	**Notes**	**Used in QoEStor**
1	8	int64	0	OCTET_DELTA_COUNT	Analog in NetFlow v9 IN_BYTES	Used
2	8	int64	0	PACKET_DELTA_COUNT	Analog in NetFlow v9 IN_PKTS	Used
4	1	int8	0	PROTOCOL_IDENTIFIER	Analog in NetFlow v9 PROTOCOL	Used
5	1	int8	0	IP_CLASS_OF_SERVICE	Analog in NetFlow v9 TOS	Used
7	2	int16	0	SOURCE_TRANSPORT_PORT	Analog in NetFlow v9 L4_SRC_PORT	Used
8	4	int32	0	SOURCE_IPV4_ADDRESS	Analog in NetFlow v9 IPV4_SRC_ADDR	Used
11	2	int16	0	DESTINATION_TRANSPORT_PORT	Analog in NetFlow v9 L4_DST_PORT	Used
12	4	int32	0	DESTINATION_IPV4_ADDRESS	Analog in NetFlow v9 IPV4_DST_ADDR	Used
16	4	int32	0	BGP_SOURCE_AS_NUMBER	Analog in NetFlow v9 SRC_AS	Used
17	4	int32	0	BGP_DESTINATION_AS_NUMBER	Analog in NetFlow v9 DST_AS	Used
152	8	int64	0	FLOW_START_MILLISECOND		Used
153	8	int64	0	FLOW_END_MILLISECOND		Used
10	2	int16	0	INPUT_SNMP	Analog in NetFlow v9 IngressInterface	Used
14	2	int16	0	OUTPUT_SNMP	Analog in NetFlow v9 EgressInterface	Used
60	1	int8	0	IP_VERSION	Analog in NetFlow v9 IP_PROTOCOL_VERSION	Used
2000	8	int64	43823	SESSION_ID		Used
2001	-	string	43823	HTTP_HOST or CN_HTTPS		Used
2002	2	int16	43823	DPI_PROTOCOL		Used
2003	-	string	43823	LOGIN	Analog in Radius User-Name	Used
225	4	int32	0	POST_NAT_SOURCE_IPV4_ADDRESS		Used
227	2	int16	0	POST_NAPT_SOURCE_TRANSPORT_PORT		Used

2010	2	int16	43823	FRGMT_DELTA_PACKS	Delta of fragmented packets.	Used
2011	2	int16	43823	REPEAT_DELTA_PACK	Delta of retransmissions.	Used
2012	4	int32	43823	PACKET_DELIVER_TIME	Delay (RTT/2) in ms (RTT=round-trip time).	Used
2016	2	int16	43823	BRIDGE_CHANNEL_NUM	Channel number (vchannel) or bridge. If vchannels are configured in DPI, the channel number will be transmitted, otherwise the bridge number.	Used
6	2	int16	0	TCP_FLAGS	TCP control bits	Used
58	2	int16	0	SRC_VLAN	VLAN ID	Used
59	2	int16	0	DST_VLAN	Post VLAN ID	Used
56	6	mac_address	0	SRC_MAC	Source MAC address	Used
57	6	mac_address	0	DST_MAC	Destination MAC address	Used
2017	-	raw	43823	MPLS Lables		Used
132	8	int64	0	DROPPED_BYTES	Delta count of dropped octets. *For example: data is dumped at minute T1 and T2. The delta will show the difference in the number of octets between minute T1 and T2.*	Used
133	8	int64	0	DROPPED_PACKETS	Delta count of dropped packets. *For example: data is dumped at minute T1 and T2. The delta will show the difference in the number of packets between minute T1 and T2.*	Used
2019	1	int8	43823	originalTOS	Original TOS value from IP header	Used
192	1	int8	0	IP_TTL	TTL packets	Used
2020	2	int16	43823	RATING_GROUP	Rating group number	Used
2021				SERVICE_FLAGS	Information about the tags received by the flow in DPI. Detected tethering is reported via IPFIX in bit 1 of the `service_flags` field. 63 bits are available for further use	Used
2022				DETECTION_FLAGS	Reserved for the detection method	Used

2023				ACTION_FLAGS	Reserved for transmitting information about operations on the flow	Used

Export Template in IPFIX Format (Netflow v10) for IPv6 Protocol

The template is similar to IPv4 except that the following fields are absent: ***SOURCE_IPV4_ADDRESS***, ***DESTINATION_IPV4_ADDRESSs***, ***POST_NAT_SOURCE_IPV4_ADDRESS***, ***POST_NAT_SOURCE_TRANSPORT_PORT***, – and the following are present:

Export Template for IPv6					
№	**Bytes**	**Data Type**	**IANA**	**Description**	**Notes**
27	16	int128	0	SOURCE_IPV6_ADDRESS	Analog in NetFlow v9 IPV6_SRC_ADDR
28	16	int128	0	DESTINATION_IPV6_ADDRESS	Analog in NetFlow v9 IPV6_DST_ADDR

Configuring NetFlow v5

In the Netflow v5 format, the original port numbers are retained in the full statistics, and information about the detected protocols is transmitted in the normally unused bytes 46–47. If you need to analyze the protocols in use, you can configure the system so that protocol information is transmitted in the port number:

```
netflow_full_port_swap=1
```

For backward compatibility with older collectors, this setting also applies to the IPFIX format; however, using it in conjunction with IPFIX is not recommended, as protocol information is transmitted in IPFIX in a separate, dedicated field.