

Export of GTP-U Information in IPFIX

draft-voyersriram-opsawg-ipfix-gtpu-02

Enabling insights in GTP forwarding plane by adding GTP-U dimensions

daniel.voyer@bell.ca
sriragop@cisco.com
thomas.graf@swisscom.com
benoit.claise@huawei.com
23 February 2024

GTP-U @ IPFIX

Data-Plane visibility is missing in GTP

- GTP is already deployed by network operators
- Data-Plane visibility is missing in GTP-U and so unable to identify the transport performance of PDU Sessions with specific QoS within a slice or within a group of slices hosted on the same User Plane Function.

3GPP TS 29.281 version 17.4.0 Release 17

19

ETSI TS 129 281 V17.4.0 (2022-10)

Octets	Bits						
	8	7	6	5	4	3	2 1
1	Version		PT	(*)	E	S	PN
2	Message Type						
3	Length (1 st Octet)						
4	Length (2 nd Octet)						
5	Tunnel Endpoint Identifier (1 st Octet)						
6	Tunnel Endpoint Identifier (2 nd Octet)						
7	Tunnel Endpoint Identifier (3 rd Octet)						
8	Tunnel Endpoint Identifier (4 th Octet)						
9	Sequence Number (1 st Octet) ^{1) 4)}						
10	Sequence Number (2 nd Octet) ^{1) 4)}						
11	N-PDU Number ^{2) 4)}						
12	Next Extension Header Type ^{3) 4)}						

NOTE 0: (*) This bit is a spare bit. It shall be sent as '0'. The receiver shall not evaluate this bit.

NOTE 1: 1) This field shall only be evaluated when indicated by the S flag set to 1.

NOTE 2: 2) This field shall only be evaluated when indicated by the PN flag set to 1.

NOTE 3: 3) This field shall only be evaluated when indicated by the E flag set to 1.

NOTE 4: 4) This field shall be present if and only if any one or more of the S, PN and E flags are set.

Figure 5.1-1: Outline of the GTP-U Header

GTP-U @ IPFIX

IPFIX entities in context of the GTP-U (1)

3GPP TS 29.281 version 17.4.0 Release 17

19

ETSI TS 129 281 V17.4.0 (2022-10)

- **gtpuFlags**

8-bit flags field defined in the GTP-U which indicates the version of GTP-U protocol, protocol type and presence of extension header, sequence number and N-PDU number in the GTP-U header.

- **gtpuMsgType**

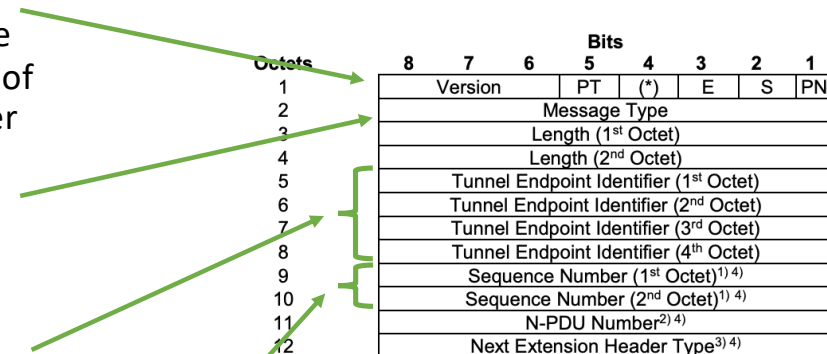
8-bit message type field defined in the GTP-U which indicates the type of GTP-U message.

- **gtpuTEid**

32-bit tunnel endpoint identifier field defined in GTP-U which unambiguously identifies a tunnel endpoint in the receiving GTP-U protocol entity for a given UDP/IP endpoint..

- **gtpuSequenceNum**

16-bit sequence number field defined in the GTP-U. This field is interpreted based on the corresponding flag value from gtpuFlags



NOTE 0: (*) This bit is a spare bit. It shall be sent as '0'. The receiver shall not evaluate this bit.
NOTE 1: 1) This field shall only be evaluated when indicated by the S flag set to 1.
NOTE 2: 2) This field shall only be evaluated when indicated by the PN flag set to 1.
NOTE 3: 3) This field shall only be evaluated when indicated by the E flag set to 1.
NOTE 4: 4) This field shall be present if and only if any one or more of the S, PN and E flags are set.

Figure 5.1-1: Outline of the GTP-U Header

GTP-U @ IPFIX

IPFIX entities in context of the GTP-U (2)

- **gtpuQFI**

8-bit QoS flow identifier field defined in PDU Session Container extension header of GTP-U. This is defined in section 5.5.3 of PDU session spec [TS.38415]. This is used to determine the QoS flow and QoS profile which are associated with the received packet..

- **gtpuPduType**

8-bit PDU type field defined in PDU Session Container extension header of GTP-U. This is defined in section 5.5.3 of PDU session spec [TS.38415]. This field indicates the structure of the PDU session UP frame..

Bits								Number of Octets
7	6	5	4	3	2	1	0	
PDU Type (=0)				QMP	SNP	MSNP	Spare	1
PPP	RQI	QoS Flow Identifier						1
PPI			Spare					0 or 1
DL Sending Time Stamp								0 or 8
DL QFI Sequence Number								0 or 3
DL MBS QFI Sequence Number								0 or 4
Padding								0-3

GTP-U @ IPFIX

Next Steps

- Data-Plane visibility is missing in GTP. **Do you recognize the problem statement?**
 - Authors believe that document should progress quickly through IETF to avoid private enterprise code points being used in GTP 5G deployments.
- > Call for adoption at OPSAWG at IETF 119**