Internet Draft
<draft-ietf-ipfix-reqs-01.txt>
Expires: August 2002

J. Quittek
NEC Europe Ltd.
T. Zseby
FhI FOKUS
B. Claise
Cisco Systems
(Editors)

February 2002

# Requirements for IP Flow Information Export
# <draft-ietf-ipfix-reqs-01.txt>

## Status of this Memo



## Abstract

This memo defines requirements for the export of measured IP flow information out of routers, traffic measurement probes and middleboxes.

## Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

Table of Content

## 1. Introduction

There are several applications that require flow-based IP traffic measurements. Such measurements could be performed by a router while forwarding the traffic, by a middlebox [MIDBOXTAX], or by a traffic measurement probe attached to a line or a monitored port. This memo defines requirements for exporting traffic flow information out of these boxes for further processing by applications located on other devices. In section 2 a selection of such applications is presented. The following sections list requirements derived from these applications.

## 2. Terminology

### 2.1. IP Traffic Flows

There are several definitions of the term 'flow' being used by the Internet community. Within this document we use the following one:

A flow is defined as a set of packets passing an observation point in the network during a certain time interval. All packets belonging to a particular flow have a set of common properties. Each property is defined as the result of applying a function to the values of:

1. one or more of packet header fields (eg. destination IP address)
2. one or more properties of the packet itself (eg. packet length)
3. one or more of fields derived from packet treatment (eg. AS number)

A packet is defined to belong to a flow if it completely satisfies all the defined properties of the flow.

This definition covers the range from a flow containing all packets observed at a network interface to a flow consisting of just a single packet between two applications with a specific sequence number. Please note that the flow definition does not match a general application-level end-to-end stream. However, an application may derive properties of application-level streams by processing measured flow data.

### 2.2. Observation Point

The observation point is a location in the network where IP packets can be observed. Examples are a line to which a probe is attached, a shared medium, such as an Ethernet-based LAN, a single port of a router, or a set of interfaces (physical or logical) of a router.

### 2.3. Metering Process

The metering process gets as input all packets observed at the observation point. On these packets it performs the actions of timestamping, sampling, filtering, mapping them to flows. Furthermore, the metering process includes maintaining flow records, computing flow statistics, and detecting flow expiration.

### 2.4. Flow Record

A flow record provides information about a specific flow that was measured at an observation point using a certain set of methods within an exporter. A flow record may contain characteristic properties of the flow, for example the source IP address, as well as measured properties of the flow, for example the total number of bytes of all packets of the flow.

### 2.5. Export Process

Flow information export denotes the process of sending flow records to one or more collectors.

### 2.6. Flow Information Collector or Collector

The collector receives flow records from one or more exporters. The collector might process or store received flow record, but these actions are out of the scope of this document.

### 2.7. IPFIX device

A device hosting at least a flow information export process. Typically, corresponding Observation points, metering processes, and exporter processes are co-located at this device, for example at a router. But also different scenarios are possible: In a hierarchical flow information collection system a collector might be co-located with the export process. Then the collector receives flow records from remote exporters and the exporter exports them to a higher level in the hierarchy.

## 3. Applications Requiring IP Flow Information Export

The following list contains a selection of applications requiring IP flow information export. Because requirements for flow export listed in further sections below are derived from these applications, their selection is crucial. The goal of this requirements document is not to cover all possible applications with all their flow export requirements, but to cover applications which are considered to be of significant importance in today's or future IP networks, and for which requirements can be met with reasonable technical effort.

Please note, that the described applications can have a large number of differing implementations. Requirement details or the weighting of requirements could differ for specific implementations. Therefore we derive the requirements from the general functionality of the selected applications. Furthermore, the list of applications should lead to a better understanding of the requirements which is particularly important when designing or implementing a traffic flow measuring device.

### 3.1 Usage-based Accounting

Several new business models for selling IP service and IP-based services are currently under investigation. Beyond flat rate services which do not need accounting, accounting for these models can be based on time or volume. Accounting data can serve as input for billing systems. Accounting can be performed per user or per user group, it can be performed just for basic IP service or individually per high-level service and/or per content type delivered. For advanced/future services, accounting may also be performed per class of service, per application, per time of day, per used (label switched) path, etc.

### 3.2 Traffic Profiling

Traffic profiling is a process of characterizing IP flows and flow aggregates by using a model that represents key parameters of the flow such as flow duration, volume, time and burstiness. It is a prerequisite for network planning, network dimensioning, trend analysis, developing business models, and other activities. It heavily depends on the particular traffic profiling objective(s) what statistics and accuracy are required from the measurements. Typical information needed for traffic profiling are the distribution of used services and protocols in the network, the amount of packets of a specific type (e.g. percentage of IPv6 packets) and specific flow profiles.

Since objectives for traffic profiling can vary, this application

requires a high flexibility of the measurement infrastructure, especially regarding the options for measurement configuration and packet classification.

### 3.3 Traffic Engineering

Traffic Engineering (TE) comprises methods for measurement, modeling, characterization and control of a network. The goal of TE is the optimization of network resource utilization and traffic performance [RFC2702]. Since control and administrative reaction to measurement results requires access to the involved network nodes, TE mechanisms and the required measurement function usually are performed within one administrative domain. Typical parameters required for TE are link utilization, load between specific network nodes, number, size and entry/exit points of the active flows and routing information.

### 3.4 Attack/Intrusion Detection

Capturing of flow information plays an important role for network security, both for detection of security violation, and for subsequent defense. In case of a Denial of Service (DOS) attack, flow monitoring can allow detection of unusual load situations or suspicious flows. In a second step, flow analysis can be performed in order to gather information about the attacking flows, and for deriving a defense strategy.

Intrusion detection is a potentially more demanding application which would not only look at specific characteristics of flows, but that may also use a stateful packet flow analysis for detecting specific, suspicious activities, or unusually frequent activities. Such activities may be characterized by specific communication patterns, detectable by characteristic sequences of certain packet types.

### 3.5 QoS Monitoring

QoS monitoring is the non-intrusive (passive) measurement of quality parameters for single flows or traffic aggregates. In contrast to intrusive (active) measurements, non-intrusive measurements utilize the existing traffic in the network for QoS analysis. Since no test traffic is sent, non-intrusive measurements can only be applied in situations where the traffic of interest is already present in the network. One example application is the validation of QoS parameters negotiated in a service level specification (SLS).

Non-intrusive measurements cannot provide the kind of controllable experiments that can be achieved with active measurements. On the other hand non-intrusive measurements do not suffer from undesired side effects caused by sending test traffic (e.g. additional load,

potential differences in treatment of test traffic and real customer traffic)

QoS monitoring often requires the correlation of data from multiple measurement instances (e.g. for measuring one-way metrics). This requires proper clock synchronization of the involved measurement instances. For some measurements packet events at the different measurement points must be correlated. For this, the provisioning of post-processing functions (e.g. the generation of packet IDs) at the measurement instances would be useful. Furthermore, QoS monitoring can lead to a huge amount of measurement result data. Therefore it would highly benefit from mechanisms to reduce the measurement data, like aggregation of results and flow sampling.

## 4. Distinguishing Flows

Packets are mapped to flows by evaluating their properties. Packets with common properties are considered to belong to the same flow. A packet showing at least one difference in the set of properties is considered to belong to a different flow.

The following subsections list a set of properties which a metering process MUST, SHOULD, or MAY be able to evaluate for mapping packets to flows. Please note that requiring the ability to evaluate a certain property does not imply that this property must be evaluated for each packet.

In other words, compliant with IPFIX means that the metering process in general must be able, via its configuration, to somehow support to distinguish flows via all the MUST fields, even if in certain circumstance/for certain applications, only a subset of the MUST fields is needed and only a subset of the MUST fields is effectively used to distinguish flows.

Which combination of properties is evaluated for a particular measurement and how these properties are evaluated depends on the configuration of the IPFIX device. The configured choice of evaluated properties strongly depends on the environment and purpose of the measurement and on the information required by the collector.

For specific deployments, only a subset of the REQUIRED properties listed below could be used to distinguish flows, in order to aggregate the flow records and reduce the number of flow records exported. On the other hand, some other deployments will require to distinguish flows by some extra parameters, like for example the TTL field of the IP header or the BGP Autonomous Systems.

### 4.1. Interfaces

The metering process MUST be able to separate flows by the incoming interface or by the outgoing interface or by both of them, if the observation point consists of one or more ports of a router.

### 4.2. IP Header Fields

The metering process MUST, SHOULD, or MAY be able to separate flows by the following fields of the IP header as indicated.

1. source IP address (MUST)
2. destination IP address (MUST)
3. transport protocol type (TCP,UDP,ICMP,...) (MUST)
4. IP version number (SHOULD)
   This requirement only applies if the device is supporting more than one version of IP.

For source address and destination address separating by full match MUST be supported as well as separation by a partial match (for example subnet masking).

### 4.3. Transport Header Fields

The metering process MUST be able to separate flows by the port numbers of the transport header in case of TCP or UDP being used as transport protocol. Both, source and destination port number MUST be supported for distinguishing flows, individually as well as in combination.

### 4.4. MPLS Label

If the metering process supports Multiprotocol Label Switching (MPLS, see [RFC3031]), then the measuring device MUST be able to separate flows by the MPLS label.

### 4.5. DiffServ Code Point

If the IPFIX device supports Differentiated Services (DiffServ) and if the observation point is local to this device, then the metering process MUST be able to separate flows by the DiffServ Code Point (DSCP, see [RFC2474]).

### 4.6. Header Compression and Encryption

If header compression or encryption is used, the metering process might not be able to access all header fields. In such a case only observation points at end points of header compression or of packet encryption are expected to meet the requirements stated in this section 4.

## 5. Metering Process

The following are requirements for the metering process. All measurements MUST be conducted from the point of view of the observation point.

### 5.1. Reliability

The metering process MUST either be reliable or missing reliability MUST be known and indicated. The metering process is reliable, if each packet passing the observation point is measured according to the configuration of the metering process. If, e.g. due to some overload, not all passing packets can be included into the metering process, then the metering process MUST be able to detect this failure and to report about it.

### 5.2. Sampling

The metering process MAY support measuring traffic by packet sampling. If sampling is supported the sampling method and its parameters MUST be well defined. If sampling parameters are changed during operation, this MUST be indicated to all collectors receiving the affected flow records.

### 5.3. Overload Behavior

In case of an overload, e. g. lack of memory or processing power, the metering process MAY change in order to cope with the lack of resources. Possible reactions include:

- Reduce the number of flow accounts. This can be achieved by more coarse grained flow measurement or by a restriction of the flow accounts to a subset of the set of original ones.
- Switch to sampling packets before they are processed by the meter or - if sampling is already performed - reduce the sampling rate.
- Stop metering.

Overload behavior is not restricted to the three options listed

above. But in any case, the overload behavior MUST be clearly defined and the collector MUST be able to distinguish the flow records exported before and after the metering process behavior change.

For example in the case of switching to sampling, the collector MUST be able to distinguish the flow records generated with sampling from the flow records generated without sampling and the sampling method and all its parameters MUST be known or indicated.

### 5.4. Timestamps

The metering process MUST be able to generate a timestamp for each observed packet. Please note that section 5.1 requires to offer reporting a timestamp for the first and the last observed packet of a flow. Therefore, the metering process MUST be able to store at least two timestamps per flow.

### 5.5. Time Synchronization

Different metering process(es) and collector(s) SHOULD be time synchronized between each other. NTP is a possible way of achieving this. Where there are so many types of synchronization between IPFIX devices and collectors, the synchronization of the devices and collectors cannot be defined as part of IPFIX. The method for time synchronization is not in the scope of IPFIX.

### 5.6. Timeout

The metering process MUST be able to detect flow timeout. A flow is considered to be timed out if no packet of this flow has been observed for a given timeout interval. The metering process MAY support means for detecting the end of a flow before a time out occurs, for example by detecting the FIN or RST bits in a TCP connection.

### 5.7. Ignore Port Copy

The metering process MAY be able to ignore packets which are generated by a port copy function acting at the same device.

## 6. Data Export

The following are requirements for exporting measured flow data out of the IPFIX device. Beside requirements on the data transfer, we separate requirements concerning the information model from requirements concerning the data model. Furthermore, we list requirements on reporting times and events and on anonymization of records.

### 6.1. Information Model

The information model for the flow information export is the list of attributes of a flow to be contained in the report (including the semantics of the attributes).

This section lists attributes an export process MUST or MAY be able to report. This does not imply that a exported flow records MUST contain all REQUIRED attributes, but that it MUST be possible to configure the device in a way that all of the REQUIRED attributes are contained in the flow records for each measured flow.

In other words, compliant with IPFIX means that the box in general must be able, via its configuration, to somehow support to report all the MUST fields, even if in certain circumstance/for certain applications, only a subset of the MUST fields is needed and only a subset of the MUST fields is effectively reported.

Beyond that, the device might offer to report also further attributes not mentioned here. A particular flow record may contain some of the "REQUIRED" attributes as well as some additional ones, for example covering future technologies.

The measuring device MUST be able to report the following attributes for each measured flow:

1. IP version number
   This requirement only applies if the device is supporting more than one version of IP.
2. source IP address
3. destination IP address
4. transport protocol type
5. source TCP/UDP port number
6. destination TCP/UDP port number
7. packet counter
   If a packet is fragmented, each fragment is counted as an individual packet.
8. byte counter
9. in case of IPv4: Type of Service

10. in case of IPv6: Flow Label
11. if BGP is supported: BGP AS#
12. if MPLS is supported: MPLS label
13. if DiffServ is supported: DSCP
14. timestamp of the first packet observed
15. timestamp of the last packet observed
16. if sampling is used: sampling method
17. if sampling is used: sampling parameters
18. unique identifier of the observation point
19. unique identifier of the export process

The measuring device MAY be able to report the following attributes for each measured flow:

20. Time To Live
21. IP header flags
22. TCP header flags
23. dropped packet counter
    If a packet is fragmented, each fragment MUST be counted as an individual packet. This requirements does not apply to probes where the value of this counter is always zero.
24. fragmented packet counter
    counter of all packets for which the fragmented bit is set in the IP header
25. multicast replication factor
    the number of outgoing packets originating from a single incoming multicast packet

### 6.2 Data Model

The data model describes how information is represented in flow records. The data model used for exporting flow information MAY be flexible concerning the flow attributes contained in flow records. A flexible record format would offer the possibility of defining records in a flexible (customizable) way regarding the number and type of contained attributes.

The data model MUST be extensible for future attributes to be added. Even if a set of attributes is fixed in the flow record, the data model MUST provide a way of extending the record by configuration or for certain implementations.

The Data Model SHOULD be independent of the underlying transport protocol, i.e. the data transfer.

### 6.3. Data Transfer

Requirements for the data transfer include reliability and security requirements. These requirements do not apply to the measuring device alone, but also to the transport network. Consequently, the export process does not necessarily have to guarantee that all requirements are met. Particularly if the security requirements are already guaranteed by the network used for data transfer, then these requirements do not have to be considered anymore by the export process. Therefore, these requirements are OPTIONAL for the export process, although they may be REQUIRED for the data transfer as specified in the appendix.

#### 6.3.1. Congestion Awareness

For the data transfer a congestion aware protocol MUST be supported.

#### 6.3.2. Reliability

Absence of reliability of the data transfer MUST be indicated covering packet loss and packet reordering.

Please note that if an unreliable transport protocol is used, reliability can be provided by higher layers. In such a case only lack of overall reliability MUST be indicated. For example reordering could be dealt with by adding a sequence number to each packet.

#### 6.3.3. Security

Confidentiality of transferred IPFIX data SHOULD be ensured.

Integrity of transferred IPFIX data MUST be ensured.

Authenticity of transferred IPFIX data MUST be ensured.

### 6.4. Push and Pull Mode Reporting

In general, there are two ways of deciding on reporting times: push mode and pull mode. In push mode, the export process decides without an external trigger on when to send a report on measured flows. In pull mode, sending a report is triggered by an explicit request from a collector. The measuring device MUST support push mode reporting, it MAY support pull mode reporting.

### 6.5. Regular Reporting Interval

The export process SHOULD be capable of reporting measured traffic data regularly according to a given interval length.

### 6.6. Notification on Specific Events

The export process MAY be capable of sending notifications to a consumer of measure data, if a specific event occurs. Such an event might be the arrival of the first packet of a new flow, or the termination of a flow after flow timeout.

### 6.7. Anonymization

The export process MAY be capable of anonymizing source and destination IP addresses in flow data before exporting them. It MAY support anonymization of port numbers and other fields. Please note that anonymization is not originally an application requirement, but derived from general requirements for treatment of traffic within a network.

## 7. Configuration

The IPFIX device MUST provide a way of configuring the traffic measurement and the traffic data export. The following parameters SHOULD be configurable:

1. specification of the observation point, e.g. a list of interfaces to be monitored.
2. specifications of flows to be measured
3. reporting data format
   Specifying the reporting data format SHOULD include a selection of attributes to be reported for each flow.
4. flow timeouts

The following parameters MAY be configurable:

5. notifications
6. sampling method and parameters, if feature is supported
7. flow anonymization, if feature is supported

If configuration is done remotely, the IPFIX device SHOULD support security of the configuration including confidentiality, integrity and authenticity. The means used for remote configuration of IPFIX devices are out of the scope of this document.

## 8. General Requirements

### 8.1. Openness

IPFIX specifications SHOULD be open to future technologies. This includes extensibility of configuration of measurement and reporting as well as extensibility of the reporting information model and data model.

### 8.2. Scalability concerning measuring devices

Data collection from hundreds of different IPFIX devices MUST be supported. The collector MUST be able to distinguish several hundred IPFIX devices by their identifiers.

### 8.3. Several Collectors

The exporting process MAY be able to export flow information to more than one collector.

## 9. Security Considerations

This document describes requirements for IP Flow Information Export (IPFIX). It therefore also states the required security features for a future IPFIX protocol. Nevertheless, the suggestion of solutions for achieving the security properties is out of scope of this document and will be part of future IPFIX documents that specify IPFIX architecture and protocol.

Like other requirements, the security requirements differ for the considered applications. The incentive to modify collected for accounting or intrusion detection for instance is usually higher than the incentive to change data collected for traffic profiling. Therefore the required security features are listed per application. Furthermore, the security requirements also differ with regard to the environment in which an IPFIX protocol is used (e.g. intra- or inter-domain). Some of these issues are part of the IPFIX architecture and with this out of scope of this document. Therefore this document also tries to consider security threats that can only occur in an insecure environment (e.g. where it can not be excluded, that an attacker might gain access to the network).

Several security hazards also occur if the IPFIX device is configured remotely (e.g. access to the measurement process, forgery of configuration information, etc.). Nevertheless, this document specifies only what parameters SHOULD or MAY be configurable for the IPFIX device. It does not deal with requirements for a protocol for

measurement configuration. Therefore security considerations regarding the measurement configuration are out of scope of this document.

The following potential security hazards for an IPFIX protocol can be identified:

- Disclosure of IP flow information data
  It may be required to keep measurement records confidential between the involved parties. Observation of IP flow information data gives an attacker information about the active flows in the network, communication endpoints and traffic patterns. This information can not only be used to spy out user behavior but also to plan and conceal future attacks. Therefore the requirements document recommends to ensure the confidentiality of the transferred data. This can be done for instance by encryption.

  Furthermore, features for anonymization may be provided by the future IPFIX protocol. With this communication endpoints can be kept confidential. Anonymization is also a useful feature to allow measurements (e.g. by a third party) without violating privacy protection.

- Forgery of exported IP flow information data
  Especially for applications like accounting or intrusion detection there are strong incentives (e.g. save money or prevent detection of an attack) to forge exported IP flow information records. This can be done either by altering data on the path or by exporting records from a device that pretends to be the IPFIX device. In order to make the IPFIX protocol resistant against such attacks this document requires to ensure authenticity and integrity of the data for the IPFIX data transfer.

  Special caution is required if security applications rely on IPFIX data With forgery of exported IP flow information data it is possible to trick on security applications. With this it can be for instance possible to pretend that a DoS attack happens without even launching a real attack.

- Denial of Service (DoS) attacks
  DoS attacks on routers or other middleboxes that have the IPFIX protocol implemented would also affect the IPFIX protocol and impair the sending of IPFIX records. Nevertheless, since such hazards are not induced specifically by the IPFIX protocol the prevention of such attacks is out of scope of this document.

IPFIX itself causes the following potential hazards for DoS attacks. It is always possible to overload the IPFIX device if it expects the reception of traffic. For IPFIX this can occur in two cases. First, if the protocol supports the pull mode (which is one option in this document) and expects requests. Secondly, if data is expected for remote measurement configuration. The first case could be prevented by ensuring authenticity for IPFIX requests. The second case should be addressed for the specification of an IPFIX remote configuration mechanism and therefore is out of scope of this document.

Also IPFIX collectors can become target of an DoS attack. This can be done by sending IPFIX data from a malicious device that pretends to be an IPFIX device. This can be prevented by ensuring authenticity of IPFIX data as stated in this document. It is also possible that collectors are flooded with IPFIX record from an authorized IPFIX devices for which the configuration was altered. Furthermore, malicious configuration or forgery of exported data can cause a loss or re-direction of flow information (e.g. if destination addresses for flow records are modified). This can lead to a disruption of upper layer services (accounting, intrusion detection, etc.) due to lack of IPFIX records. This can be prevented by controlling configuration access and by ensuring the integrity of exported data.

## 10. Acknowledgments

We like to thank all the people contributing to the requirements discussion on the mailing list for a lot of valuable comments.

## 11. References

[MIDBOXTAX] B. Carpenter, "Middleboxes: taxonomy and issues", RFC 3234, February 2002.

[RFC3031] E. Rosen, A. Viswanathan, R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, January 2001.

[RFC2474] K. Nichols, S. Blake, F. Baker, D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.

[RFC2702] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, J. McManus, "Requirements for Traffic Engineering Over MPLS", RFC 2702, September 1999.

[RFC2274] U. Blumenthal, B. Wijnen "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3), RFC 2274, January 1998.

## 12. List of Authors

Juergen Quittek
NEC Europe Ltd., Network Laboratories
Adenauerplatz 6
69115 Heidelberg
Germany

Phone: +49 6221 90511-15
EMail: quittek@ccrle.nec.de

Tanja Zseby
Fraunhofer Institute for Open Communication Systems (FOKUS)
Kaiserin-Augusta-Allee 31
10589 Berlin
Germany

Phone: +49 30 3463 7153
Email: zseby@fokus.fhg.de

Georg Carle
Fraunhofer Institute for Open Communication Systems (FOKUS)
Kaiserin-Augusta-Allee 31
10589 Berlin
Germany

Phone: +49 30 3463 7149
Email: carle@fokus.fhg.de

Sebastian Zander
Fraunhofer Institute for Open Communication Systems (FOKUS)
Kaiserin-Augusta-Allee 31
10589 Berlin
Germany

Phone: +49 30 3463 7287
Email: zander@fokus.fhg.de

Benoit Claise
Cisco Systems
De Kleetlaan 6a b1
1831 Diegem
Belgium

Phone: +32 2 704 5622
Email: bclaise@cisco.com

K.C. Norseth
Enterasys Networks
2691 S. Decker Lake Lane
Salt Lake City, Utah 84119
USA

Phone: +1 801 887 9823
Email: knorseth@enterasys.com

Appendix: Derivation of Requirements form Target Applications

The following table documents, how the requirements stated in sections 3-7 are derived from requirements of the applications listed in section 2.

Used abbreviations:
M = MUST
S = SHOULD
O = MAY (OPTIONAL)
- = DONT CARE

IPFIX
E: QoS Monitoring
D: Attack/Intrusion Detection
C: Traffic Engineering
B: Traffic Profiling
A: Usage-based Accounting

| Sect. | Requirement | A | B | C | D | E | IPFIX |
|---|---|---|---|---|---|---|---|
| 4. | DISTINGUISHING FLOWS | | | | | | |
| 4. | Combination of required attributes | M | M | M | M | M | M |
| 4.1. | in/out IF | S | M | M | S | S | M |
| 4.2. | src/dst address | M | M | M | M | M | M |
| 4.2. | Masking of IP adresses | M | M | M | M | M | M |
| 4.2. | transport protocol | M | M | - | M | M | M |
| 4.2. | version field | - | S | S (b) | O | O | S |
| 4.3. | src/dst port | M | M | - | M | M | M |

| Sect. | Requirement | A | B | C | D | E | IPFIX |
|---|---|---|---|---|---|---|---|
| 4.4. | MPLS label (a) | S | S | M (c) | O | S | M |
| 4.5. | DSCP (a) | M | S | M | O | M | M |
| 5. | METERING PROCESS | | | | | | |
| 5.1. | Reliability | M | S | S | S | S | M |
| 5.1. | Indication of missing reliability | - | M | M | M | M | |
| 5.2. | Sampling (g) | O | O | O | O | O | O |
| 5.2. | Dynamic sampling | O | O | O | O | O | O |
| 5.4. | Timestamping at measurement device | M | O | O | S | M | M |
| 5.5. | Time synchronization | S | S | S | S | S | S |
| 5.6. | Flow timeout | M (d) | S | - | O | O | M |
| 5.7. | Ignore port copy | O | O | O | O | O | O |
| 6. | DATA EXPORT | | | | | | |
| 6.1. | INFORMATION MODEL | | | | | | |
| 6.1. | IP Version | - | M | M | O | O | M |
| 6.1. | src/dst address | M | M | M | M | M | M |
| 6.1. | transport protocol | M | M | - | M | M | M |
| 6.1. | src/dst transport | M | M | - | M | M | M |
| 6.1. | Packet counter (h) | S | M | M | S | S | M |
| 6.1. | Byte counter | M | M | M | S | S | M |

| Sect. | Requirement | A | B | C | D | E | IPFIX |
|---|---|---|---|---|---|---|---|
| 6.1. | Dropped Packet Counter (h,i) | O | M | M | S | M | M |
| 6.1. | ToS Byte | M | S | M | O | M | M |
| 6.1. | Flow Label | M | S | M | O | M | M |
| 6.1. | BGP AS# | - | S | M | - | - | M |
| 6.1. | MPLS label (a) | S | S | M (c) | O | S | M |
| 6.1. | DSCP (a) | M | S | M | O | M | M |
| 6.1. | Timestamps for first/last packet | M | O | O | S | S | M |
| 6.1. | Sampling methods & parameters (k) | M | M | M | M | M | M |
| 6.1. | observation point identifier | M | M | M | M | M | M |
| 6.1. | measuring device identity | M | M | M | M | M | M |
| 6.1. | TTL | O | O | O | O | O | O |
| 6.1. | IP header flags | - | O | O | O | O | O |
| 6.1. | TCP header flags | - | O | O | O | - | O |
| 6.1. | Fragment counter | - | O | O | O | O | O |
| 6.1. | Multicast replication factor | O | O | O | - | - | O |
| 6.1. | Flow configuration | O | O | O | O | O | O |
| 6.2. | DATA MODEL | | | | | | |
| 6.2. | Flexibility | O | O | O | O | O | O |
| 6.2. | Extensibility | M | M | M | M | M | M |

| Sect. | Requirement | A | B | C | D | E | IPFIX |
|---|---|---|---|---|---|---|---|
| 6.3. | DATA TRANSFER | | | | | | |
| 6.3.1. | Congestion aware | M | M | M | M | M | M |
| 6.3.2. | Reliability | M | S | S | S | S | M |
| 6.3.3. | Confidentiality | S | S | S | S | S | S |
| 6.3.4. | Integrity | M | M | M | M | M | M |
| 6.3.5. | Authenticity | M | M | M | M | M | M |
| 6.4. | REPORTING TIMES | | | | | | |
| 6.4. | Push mode | M | O (e) | O (e) | M | S (e,f) | M |
| 6.4. | Pull mode | O | O (e) | O (e) | O | O (e) | O |
| 6.4.1. | Regular Interval | S | S | S | S | S | S |
| 6.6. | Notifications | O | O | O | O | O | O |
| 6.7. | Anonymization | O | O | O | O | O | O |
| 7. | CONFIGURATION | | | | | | |
| 7. | Config Measurement & Data Export | M | M | M | M | M | M |
| 7. | Config Observation Point | S | S | S | S | S | S |
| 7. | Config Flow Specifications | S | S | S | S | S | S |
| 7. | Config Report Data Format | S | S | S | S | S | S |
| 7. | Config Flow Timeouts | S | S | S | S | O | S |

| Sect. | Requirement | A | B | C | D | E | IPFIX |
|---|---|---|---|---|---|---|---|
| 7. | Config Notifications | O | O | O | O | O | O |
| 7. | Config Sampling | O | O | O | O | O | O |
| 7. | Config Anonymization | O | O | O | O | O | O |
| 7. | Config Security | O | O | O | O | O | O |
| 8. | GENERAL REQUIREMENTS | | | | | | |
| 8.1. | Openness | S | S | S | S | S | S |
| 8.2. | Scalability: data collection from hundreds of measurement devices | M | S | M | O | S | M |
| 8.3. | Several Collectors | O | O | O | O | O | O |

Remarks:

(a) If feature is supported.

(b) The differentiation of IPv4 and IPv6 is for TE of importance. So we tended to make this a MUST. Nevertheless, a SHOULD seems to be sufficient to perform most TE tasks and allows us to have a SHOULD for IPFIX instead of a MUST.

(c) For TE in an MPLS network the label is essential. Therefore a MUST is given here leading to a MUST in IPFIX.

(d) Precise time-based accounting requires reaction to a flow timeout.

(e) Either push or pull has to be supported.

(f) Required, in order to immediately report drop indications for SLA validation.

(g) If sampling is supported the parameters and methods MUST be well defined.

(h) If a packet is fragmented, each fragment is counted as an individual packet.

(i) Only if measurement is done on data path i.e. has access to forwarding decision.

(k) If sampling is used.