

Internet Engineering Task Force (IETF)
Request for Comments: 6615
Obsoletes: 5815
Category: Standards Track
ISSN: 2070-1721

T. Dietz, Ed.
NEC Europe Ltd.
A. Kobayashi
NTT PF Labs
B. Claise
Cisco Systems, Inc.
G. Muenz
Technische Universitaet Muenchen
June 2012

Definitions of Managed Objects for IP Flow Information Export

Abstract

This document defines managed objects for IP Flow Information eXport (IPFIX). These objects provide information for monitoring IPFIX Exporters and IPFIX Collectors, including basic configuration information.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <http://www.rfc-editor.org/info/rfc6615>.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. IPFIX Documents Overview	3
3. The Internet-Standard Management Framework	4
4. Terminology	4
5. Structure of the IPFIX MIB	4
5.1. The Transport Session Table	4
5.2. The Template Table	7
5.3. The Template Definition Table	9
5.4. The Export Table	11
5.5. The Metering Process Table	13
5.6. The Observation Point Table	14
5.7. The Selection Process Table	15
5.8. The Statistical Tables	16
5.8.1. The Transport Session Statistical Table	16
5.8.2. The Template Statistical Table	16
5.8.3. The Metering Process Statistical Table	16
5.8.4. The Selection Process Statistical Table	16
6. Structure of the IPFIX SELECTOR MIB	16
6.1. The Selector Functions	17
7. Relationship to Other MIB Modules	19
7.1. Relationship to the ENTITY MIB and Interfaces MIB	19
7.2. MIB Modules Required for IMPORTS	19
8. MIB Definitions	20
8.1. IPFIX MIB Definition	20
8.2. IPFIX SELECTOR MIB Definition	56
9. Security Considerations	60
10. IANA Considerations	61
11. Acknowledgments	62
12. References	62
12.1. Normative References	62
12.2. Informative References	63

1. Introduction

This document defines two MIB modules for monitoring IP Flow Information eXport (IPFIX) Devices, including Exporters and Collectors. While most of the objects defined by the IPFIX MIB module must be implemented, some objects may be implemented corresponding to the functionality implemented in the equipment. Since the IPFIX architecture [RFC5470] foresees the possibility of using Filtering and/or Sampling functions to reduce the data volume, this document also provides the IPFIX SELECTOR MIB module, which contains the standardized selection methods and is controlled by IANA. The full configuration of the IPFIX Metering Process is out of the scope of these MIB modules.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. IPFIX Documents Overview

The IPFIX protocol provides network administrators with access to IP Flow information. The architecture for the export of measured IP Flow information out of an IPFIX Exporting Process to a Collecting Process is defined in [RFC5470], per the requirements defined in [RFC3917]. The protocol document [RFC5101] specifies how IPFIX Data Records and Templates are carried via a congestion-aware transport protocol from IPFIX Exporting Processes to IPFIX Collecting Processes. IPFIX has a formal description of IPFIX Information Elements -- their name, type, and additional semantic information -- as specified in [RFC5102]. Finally, [RFC5472] describes what type of applications can use the IPFIX protocol and how they can use the information provided. It furthermore shows how the IPFIX framework relates to other architectures and frameworks.

It is assumed that Flow metering, export, and collection are performed according to the IPFIX architecture defined in [RFC5470]. The monitored configuration parameters of the export and collection of Flow Templates and Data Records are modeled according to [RFC5101]. Packet selection methods that may be optionally used by the IPFIX Metering Process are not considered in this MIB document. They are defined in the Packet Sampling (PSAMP) framework [RFC5474] and Sampling techniques [RFC5475] documents. Nevertheless, the basis for defining Sampling and Filtering functions is given with the IPFIX SELECTOR MIB module. Since the PSAMP export protocol [RFC5476] is based on the IPFIX protocol, the Sampling and Filtering functions can be added to the IPFIX SELECTOR MIB module as needed.

3. The Internet-Standard Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to section 7 of RFC 3410 [RFC3410].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies MIB modules that are compliant to the SMIV2, which is described in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580 [RFC2580].

4. Terminology

The definitions of basic terms such as IP Traffic Flow, Exporting Process, Collecting Process, Observation Points, etc. can be found in the IPFIX protocol document [RFC5101].

5. Structure of the IPFIX MIB

The IPFIX MIB module consists of seven main tables: the Transport Session table, the Template table and the corresponding Template Definition table, the Export table, the Metering Process table, the Observation Point table, and the Selection Process table. Since the IPFIX architecture [RFC5470] foresees the possibility of using Filtering and/or Sampling functions to reduce the data volume, the IPFIX MIB module provides the basic objects for these functions with the Selection Process table. The IPFIX SELECTOR MIB module, defined in the next section, provides the standard Filtering and Sampling functions that can be referenced in the `ipfixSelectionProcessTable`.

All remaining objects contain statistical values for the different tables contained in the MIB module.

The following subsections describe all tables in the IPFIX MIB module.

5.1. The Transport Session Table

The Transport Session is the basis of the MIB module. The Transport Session table (`ipfixTransportSessionTable`) contains all Transport Sessions between the Exporter and Collector. The table specifies the transport layer protocol of the Transport Session and, depending on that protocol, further parameters for the Transport Session. In the case of UDP and TCP, these are the source and destination address as

well as the source and destination port. For the Stream Control Transmission Protocol (SCTP), the table contains `ipfixTransportSessionSctpAssocId`, which is the index for the SCTP association in the SCTP MIB module [RFC3873]. The mode of operation of the device, i.e., whether the Transport Session is used for collecting or exporting, is given in the `ipfixTransportSessionDeviceMode` object. Further on, the table contains the configured refresh parameters for Templates and Options Templates that are used across unreliable connections such as UDP. Finally, the IPFIX version that is exported or collected by this Transport Session and a status of the Transport Session are given in the table.

To illustrate the use of this table, let us assume the following scenario: we have an Exporter on IP address 192.0.2.22 and a Collector on IP address 192.0.2.37. The Exporter uses TCP to export Templates and Data Records. The same Exporter also exports, with UDP, to a Collector with the IP address of 192.0.2.44. This would lead to the following Transport Session table on the Exporter:

```

ipfixTransportSessionTable (1)
|
+-- ipfixTransportSessionEntry (1)
|
|   +- index (5) (ipfixTransportSessionIndex)
|   |   +- ipfixTransportSessionIndex (1) = 5
|   |   +- ipfixTransportSessionProtocol (2) = 6 (TCP)
|   |   +- ipfixTransportSessionSourceAddressType (3) = 1 (ipv4)
|   |   +- ipfixTransportSessionSourceAddress (4) = 192.0.2.22
|   |   +- ipfixTransportSessionDestinationAddressType (5) = 1 (ipv4)
|   |   +- ipfixTransportSessionDestinationAddress (6) = 192.0.2.37
|   |   +- ipfixTransportSessionSourcePort (7) = 7653
|   |   +- ipfixTransportSessionDestinationPort (8) = 4739
|   |   +- ipfixTransportSessionSctpAssocId (9) = 0
|   |   +- ipfixTransportSessionDeviceMode (10) = exporting(1)
|   |   +- ipfixTransportSessionTemplateRefreshTimeout (11) = 0
|   |   +- ipfixTransportSessionOptionsTemplateRefreshTimeout (12) = 0
|   |   +- ipfixTransportSessionTemplateRefreshPacket (13) = 0
|   |   +- ipfixTransportSessionOptionsTemplateRefreshPacket (14) = 0
|   |   +- ipfixTransportSessionIpfixVersion (15) = 10
|   |   +- ipfixTransportSessionStatus (16) = 2 (active)
|   |
|   |
|   |
|   +- index (11) (ipfixTransportSessionIndex)
|   |   +- ipfixTransportSessionIndex (1) = 11
|   |   +- ipfixTransportSessionProtocol (2) = 17 (UDP)
|   |   +- ipfixTransportSessionSourceAddressType (3) = 1 (ipv4)
|   |   +- ipfixTransportSessionSourceAddress (4) = 192.0.2.22
|   |   +- ipfixTransportSessionDestinationAddressType (5) = 1 (ipv4)
|   |   +- ipfixTransportSessionDestinationAddress (6) = 192.0.2.44
|   |   +- ipfixTransportSessionSourcePort (7) = 14287
|   |   +- ipfixTransportSessionDestinationPort (8) = 4739
|   |   +- ipfixTransportSessionSctpAssocId (9) = 0
|   |   +- ipfixTransportSessionDeviceMode (10) = exporting(1)
|   |   +- ipfixTransportSessionTemplateRefreshTimeout (11) = 100
|   |   +- ipfixTransportSessionOptionsTemplateRefreshTimeout (12)
|   |   |   = 100
|   |   +- ipfixTransportSessionTemplateRefreshPacket (13) = 10
|   |   +- ipfixTransportSessionOptionsTemplateRefreshPacket (14) = 10
|   |   +- ipfixTransportSessionIpfixVersion (15) = 10
|   |   +- ipfixTransportSessionStatus (16) = 2 (active)

```

The values in parentheses are the OID numbers. The Collectors would then have the same entry, except that the index would most likely differ and the ipfixTransportSessionDeviceMode value would be collecting(2).

5.2. The Template Table

The Template table lists all Templates (including Options Templates) that are sent (by an Exporter) or received (by a Collector). The (Options) Templates are unique per Observation Domain and per Transport Session. Note that the Transport Session also gives the device mode, i.e., Exporter or Collector. Thus, the table is indexed by

- o the Transport Session Index (`ipfixTransportSessionIndex`) and
- o the Observation Domain ID (`ipfixTemplateObservationDomainId`).

It contains the Set ID and an access time denoting the time when the (Options) Template was last sent or received.

To resume the above example, the Exporter may want to export a Template and an Options Template for each Transport Session defined above. This leads to the following Template table, which defines the Template and Options Template:

```

ipfixTemplateTable (3)
|
+-- ipfixTemplateEntry (1)
|
|   +- index (5) (ipfixTransportSessionIndex)
|   |   +- index (3) (ipfixTemplateObservationDomainId)
|   |   |   + index (257) (ipfixTemplateId)
|   |   |   |   +- ipfixTemplateObservationDomainId (1) = 3
|   |   |   |   +- ipfixTemplateId (2) = 257
|   |   |   |   +- ipfixTemplateSetId (3) = 2
|   |   |   |   +- ipfixTemplateAccessTime (4)
|   |   |   |       = 2008-7-1,12:49:11.2,+2:0
|   |   |   + index (264) (ipfixTemplateId)
|   |   |   |   +- ipfixTemplateObservationDomainId (1) = 3
|   |   |   |   +- ipfixTemplateId (2) = 264
|   |   |   |   +- ipfixTemplateSetId (3) = 3
|   |   |   |   +- ipfixTemplateAccessTime (4)
|   |   |   |       = 2008-7-1,12:47:04.8,+2:0
|   |   |   .
|   |   |   .
|   |   |   .
|   |   +- index (11) (ipfixTransportSessionIndex)
|   |   |   +- index (3) (ipfixTemplateObservationDomainId)
|   |   |   |   + index (273) (ipfixTemplateId)
|   |   |   |   |   +- ipfixTemplateObservationDomainId (1) = 3
|   |   |   |   |   +- ipfixTemplateId (2) = 273
|   |   |   |   |   +- ipfixTemplateSetId (3) = 2
|   |   |   |   |   +- ipfixTemplateAccessTime (4)
|   |   |   |   |       = 2008-7-1,12:49:11.2,+2:0
|   |   |   |   + index (289) (ipfixTemplateId)
|   |   |   |   |   +- ipfixTemplateObservationDomainId (1) = 3
|   |   |   |   |   +- ipfixTemplateId (2) = 289
|   |   |   |   |   +- ipfixTemplateSetId (3) = 3
|   |   |   |   |   +- ipfixTemplateAccessTime (4)
|   |   |   |   |       = 2008-7-1,12:47:04.8,+2:0

```

We assume that the Transport Session that is stored with index 5 in the Transport Session table of the Exporter is stored with index 17 in the Transport Session table of the (corresponding) Collector. Then, the Template table would look as follows:


```

ipfixTemplateTable (3)
|
+-- ipfixTemplateEntry (1)
|
|   +- index (17) (ipfixTransportSessionIndex)
|   +- index (3) (ipfixTemplateObservationDomainId)
|   + index (257) (ipfixTemplateId)
|   |   +- ipfixTemplateObservationDomainId (1) = 3
|   |   +- ipfixTemplateId (2) = 257
|   |   +- ipfixTemplateSetId (3) = 2
|   |   +- ipfixTemplateAccessTime (4)
|   |   |   = 2008-7-1,12:49:11.8,+2:0
|   |
|   + index (264) (ipfixTemplateId)
|   |   +- ipfixTemplateObservationDomainId (1) = 3
|   |   +- ipfixTemplateId (2) = 264
|   |   +- ipfixTemplateSetId (3) = 3
|   |   +- ipfixTemplateAccessTime (4)
|   |   |   = 2008-7-1,12:47:05.3,+2:0

```

The table on the second Collector would be analogous to the one shown above.

5.3. The Template Definition Table

The Template Definition table lists all the Information Elements contained in a Template or Options Template. Therefore, it has the same indexes as the corresponding Template table plus the Template ID. Its own index denotes the order of the Information Element inside the Template. Besides the Information Element ID and the length of the encoded value, the table contains the enterprise number for enterprise-specific Information Elements and flags for each Information Element. The flags indicate whether the Information Element is used for scoping or as a Flow Key.

To resume the above example again, the Exporter is configured to export the octets received and dropped at the Observation Point since the last export of these values. In addition, it exports the start and end time of the Flow relative to the timestamp contained in the IPFIX header. This leads to the following Template Definition table on the Exporter:

```

ipfixTemplateDefinitionTable (4)
|
+-- ipfixTemplateDefinitionEntry (1)
|
|   +- index (5) (ipfixTransportSessionIndex)
|   +- index (3) (ipfixTemplateObservationDomainId)
|   + index (257) (ipfixTemplateId)
|   +- index (1) (ipfixTemplateDefinitionIndex)
|   |   +- ipfixTemplateDefinitionIndex (1) = 1
|   |   +- ipfixTemplateDefinitionIeId (2) = 158
|   |   |   (flowStartDeltaMicroseconds)
|   |   +- ipfixTemplateDefinitionIeLength (3) = 4
|   |   +- ipfixTemplateDefinitionEnterpriseNumber (4) = 0
|   |   +- ipfixTemplateDefinitionFlags (5) = 0
|   |
|   +- index (2) (ipfixTemplateDefinitionIndex)
|   |   +- ipfixTemplateDefinitionIndex (1) = 2
|   |   +- ipfixTemplateDefinitionIeId (2) = 159
|   |   |   (flowEndDeltaMicroseconds)
|   |   +- ipfixTemplateDefinitionIeLength (3) = 4
|   |   +- ipfixTemplateDefinitionEnterpriseNumber (4) = 0
|   |   +- ipfixTemplateDefinitionFlags (5) = 0
|   |
|   +- index (3) (ipfixTemplateDefinitionIndex)
|   |   +- ipfixTemplateDefinitionIndex (1) = 3
|   |   +- ipfixTemplateDefinitionIeId (2) = 1
|   |   |   (octetDeltaCount)
|   |   +- ipfixTemplateDefinitionIeLength (3) = 8
|   |   +- ipfixTemplateDefinitionEnterpriseNumber (4) = 0
|   |   +- ipfixTemplateDefinitionFlags (5) = 0
|   |
|   +- index (4) (ipfixTemplateDefinitionIndex)
|   |   +- ipfixTemplateDefinitionIndex (1) = 4
|   |   +- ipfixTemplateDefinitionIeId (2) = 132
|   |   |   (droppedOctetDeltaCount)
|   |   +- ipfixTemplateDefinitionIeLength (3) = 8
|   |   +- ipfixTemplateDefinitionEnterpriseNumber (4) = 0
|   |   +- ipfixTemplateDefinitionFlags (5) = 0

```

The corresponding table entry on the Collector is the same, except that it would have another `ipfixTransportSessionIndex`, e.g., 17 as in the previous example.

5.4. The Export Table

On Exporters, the Export table (`ipfixExportTable`) can be used to support features like failover, load-balancing, duplicate export to several Collectors, etc. The table has three indexes that link an entry with

- o the Metering Process table (`ipfixMeteringProcessCacheId`; see below) and
- o the Transport Session table (`ipfixTransportSessionIndex`).

Those entries with the same `ipfixExportIndex` and the same `ipfixMeteringProcessCacheId` define a Transport Session group. The member type for each group member describes its functionality. All Transport Sessions referenced in this table **MUST** have a `ipfixTransportSessionDeviceMode` value of `exporting(1)`.

If the Exporter does not use Transport Session grouping, then each `ipfixExportIndex` contains a single `ipfixMeteringProcessCacheId`, and thus a single Transport Session (`ipfixTransportSessionIndex`); this session **MUST** have a member type value of `primary(1)`.

For failover, a Transport Session group can contain one Transport Session with member type `primary(1)` and several Transport Sessions with type `secondary(2)`. Entries with other member types are not allowed for that type of group. For load-balancing or parallel export, all Transport Sessions in the group **MUST** have the same member type -- either `loadBalancing(4)` or `parallel(3)`.

The algorithms used for failover or load-balancing are out of the scope of this document.

To continue the example, we assume that the Exporter uses the two connections shown in the examples above as one primary Transport Session protected by a secondary Transport Session. The Exporter then has the following entries in the `ipfixExportTable`:

```

ipfixExportTable (5)
|
+-- ipfixExportEntry (1)
|
|   +- index (7) (ipfixExportIndex)
|   |   +- index (9) (ipfixMeteringProcessCacheId)
|   |   |   +- index (5) (ipfixTransportSessionIndex)
|   |   |   |   +- ipfixExportIndex (1) = 7
|   |   |   |   +- ipfixExportMemberType (2) = 1 (primary)
|   |   |   +- index (11) (ipfixTransportSessionIndex)
|   |   |   |   +- ipfixExportIndex (1) = 7
|   |   |   |   +- ipfixExportMemberType (2) = 2 (secondary)
|   |   +- index (8) (ipfixExportIndex)
|   |   +- index (9) (ipfixMeteringProcessCacheId)
|   |   +- index (5) (ipfixTransportSessionIndex)
|   |   |   +- ipfixExportIndex (1) = 8
|   |   |   +- ipfixExportMemberType (2) = 2 (secondary)
|   |   +- index (11) (ipfixTransportSessionIndex)
|   |   |   +- ipfixExportIndex (1) = 8
|   |   |   +- ipfixExportMemberType (2) = 1 (primary)

```

The example shows that the Exporter uses the Metering Process cache (index (9)), explained below, to export IPFIX Data Records for Transport Sessions 5 and 11. Templates 257 and 264 defined above are exported within Transport Session 5 as primary, while the secondary Transport Session is 11. Templates 273 and 289 are exported within Transport Session 11 as primary, while the secondary Transport Session is 5.

Here are the steps required by a manager in order to understand what the backups are (if any) for Template Records exported from a specific Exporter to a specific Collector:

1. Look up the Collector IP address in the `ipfixTransportSessionDestinationAddress` object (in the `ipfixTransportSessionTable`).
2. From the same row, double-check the Exporter IP address in the `ipfixTransportSessionSourceAddress` object.
3. From the same row, write down the `ipfixTransportSessionIndex` value.

4. Use that `ipfixTransportSessionIndex` value in the `ipfixTemplateTable` and look up the pairs of (`ipfixTemplateObservationDomainId`, `ipfixTemplateId`). From there, the manager deduces the Template Record(s) (`ipfixTemplateId`), exported from the Observation Domain(s) (`ipfixTemplateObservationDomainId`) on the tracked Exporter (`ipfixTransportSessionSourceAddress`) to the tracked Collector (`ipfixTransportSessionDestinationAddress`).
5. Reusing the same `ipfixTransportSessionIndex` in the `ipfixExportTable`, look in the table for a value of `ipfixExportMemberType` that equals "primary". Note that there could be multiple entries for which the `ipfixExportMemberType` equals "primary" in the `ipfixExportTable`, so multiple iterations might be required until the correct value of `ipfixTransportSessionIndex` is found.
6. From the same row, write down the `ipfixExportIndex` value.
7. In the `ipfixExportTable`, under the same three index values (`ipfixExportIndex`, `ipfixMeteringProcessCacheId`, and `ipfixTransportSessionIndex`), look up the entries for which `ipfixExportMemberType` is different than "primary". Write down the associated `ipfixTransportSessionIndex` value.
8. From the `ipfixTransportSessionTable`, look up the Transport Session details for this `ipfixTransportSessionIndex` value -- for example, the secondary Collector IP address and port (`ipfixTransportSessionDestinationAddress` and `ipfixTransportSessionSourcePort`).

5.5. The Metering Process Table

The Metering Process, as defined in [RFC5101], consists of a set of functions. Maintaining the Flow Records is one of them. This function is responsible for passing the Flow Records to the Exporting Process and also for detecting Flow expiration. The Flow Records that are maintained by the Metering Process can be grouped by the Observation Points at which they are observed. The instance that maintains such a group of Flow Records is a kind of cache. For this reason, the Metering Process table (`ipfixMeteringProcessTable`) is indexed by cache IDs (`ipfixMeteringProcessCacheId`). Each cache can be maintained by a separate instance of the Metering Process. To specify the Observation Point(s) where the Flow Records are gathered, the `ipfixMeteringProcessObservationPointGroupRef` may contain an `ipfixObservationPointGroupId` from the Observation Point table (`ipfixObservationPointTable`), which is described in the next subsection. If an Observation Point is not specified for the Flow

Records, the `ipfixMeteringProcessObservationPointGroupRef` MUST be zero(0). The timeouts (`ipfixMeteringProcessCacheActiveTimeout` and `ipfixMeteringProcessCacheIdleTimeout`) specify when Flows are expired.

```

ipfixMeteringProcessTable (6)
|
+-- ipfixMeteringProcessEntry (1)
   |
   +- index (9) (ipfixMeteringProcessCacheId)
      +- ipfixMeteringProcessCacheId (1) = 9
      +- ipfixMeteringProcessObservationPointGroupRef (2) = 17
      +- ipfixMeteringProcessCacheActiveTimeout (3) = 100
      +- ipfixMeteringProcessCacheIdleTimeout (4) = 100

```

5.6. The Observation Point Table

The Observation Point table (`ipfixObservationPointTable`) groups Observation Points with the `ipfixObservationPointGroupId`. Each entry contains the Observation Domain ID in which the Observation Point is located and a reference to the ENTITY MIB module [RFC4133] or the Interfaces MIB module [RFC2863]. The objects in the ENTITY MIB module referenced by `ipfixObservationPointPhysicalEntity`, or the objects in the Interfaces MIB module referenced by `ipfixObservationPointPhysicalInterface`, denote the Observation Point. At least one reference for the objects `ipfixObservationPointPhysicalEntity` or `ipfixObservationPointPhysicalInterface` MUST exist for a valid Observation Point entry. If a reference to the Observation Point is given in both object `ipfixObservationPointPhysicalEntity` and `ipfixObservationPointPhysicalInterface`, then both MUST point to the same physical interface. However, if one of two references (`ipfixObservationPointPhysicalEntity` or `ipfixObservationPointPhysicalInterface`) cannot be given, its reference MUST be 0. In addition, a direction can be given to render more specifically which Flow to monitor.

```

ipfixObservationPointTable (7)
|
+-- ipfixObservationPointEntry (1)
|
|   +- index (17) (ipfixObservationPointGroupId)
|   |   +- index (1) (ipfixObservationPointIndex)
|   |   |   +- ipfixObservationPointGroupId (1) = 17
|   |   |   +- ipfixObservationPointIndex (2) = 1
|   |   |   +- ipfixObservationPointObservationDomainId (3) = 3
|   |   |   +- ipfixObservationPointPhysicalEntity (4) = 6
|   |   |   +- ipfixObservationPointPhysicalInterface (5) = 0
|   |   |   +- ipfixObservationPointPhysicalEntityDirection (6)
|   |   |   |   = 3 (both)
|   |
|   +- index (2) (ipfixObservationPointIndex)
|   |   +- ipfixObservationPointGroupId (1) = 17
|   |   +- ipfixObservationPointIndex (2) = 2
|   |   +- ipfixObservationPointObservationDomainId (3) = 3
|   |   +- ipfixObservationPointPhysicalEntity (4) = 0
|   |   +- ipfixObservationPointPhysicalInterface (5) = 0
|   |   +- ipfixObservationPointPhysicalEntityDirection (6)
|   |   |   = 1 (ingress)

```

5.7. The Selection Process Table

This table supports the usage of Filtering and Sampling functions, as described in [RFC5470]. It contains lists of functions per Metering Process cache (ipfixMeteringProcessCacheId). The selection process index ipfixSelectionProcessIndex forms groups of selection methods that are applied to an observed packet stream. The selection process selector index (ipfixSelectionProcessSelectorIndex) indicates the order in which the functions are applied to the packets observed at the Observation Points associated with the Metering Process cache. The selection methods are applied in increasing order; i.e., selection methods with a lower ipfixSelectionProcessSelectorIndex are applied first. The functions are referenced by object identifiers pointing to each function with its parameters. If the selection method does not use parameters, then it MUST point to the root of the function subtree (see also Section 6). If the function uses parameters, then it MUST point to an entry in the parameter table of the selection method. If no Filtering or Sampling function is used for a Metering Process, then an entry for the Metering Process SHOULD be created that points to the Select All function (ipfixFuncSelectAll).

5.8. The Statistical Tables

Statistical tables that augment the `ipfixTransportSessionTable`, `ipfixTemplateTable`, `ipfixMeteringProcessTable`, and `ipfixSelectionProcessTable` have been defined. All the statistical tables contain a discontinuity object that holds a timestamp denoting the time when a discontinuity event occurred, in order to notify the management system that the counters contained in those tables might not be continuous anymore.

5.8.1. The Transport Session Statistical Table

The Transport Session Statistical table (`ipfixTransportSessionStatsTable`) augments the `ipfixTransportSessionTable` with statistical values. It contains the rate (in bytes per second) at which it receives or sends out IPFIX Messages; the number of bytes, packets, messages, Records, Templates, and Options Templates received or sent; and the number of messages that were discarded.

5.8.2. The Template Statistical Table

This table contains a statistical value for each Template. It augments the Template table (`ipfixTemplateTable`) and specifies the number of Data Records exported or collected for the Template.

5.8.3. The Metering Process Statistical Table

This table augments the Metering Process table (`ipfixMeteringProcessTable`). It contains the statistical values for the exported Data Records and the number of unused cache entries.

5.8.4. The Selection Process Statistical Table

This table augments the Selection Process table (`ipfixSelectionProcessTable`) and introduces two generic statistical values: the number of packets observed and the number of packets dropped by the selection method.

6. Structure of the IPFIX SELECTOR MIB

The IPFIX SELECTOR MIB module defined in this section provides the standard Filtering and Sampling functions that can be referenced in the `ipfixSelectionProcessTable`. All standard Filtering and Sampling functions MUST be registered in the subtree under object `ipfixSelectorFunctions` (`iso.org.dod.internet.mgmt.mib-2.ipfixSelectorMIB.ipfixSelectorObjects.ipfixSelectorFunctions`, or `1.3.6.1.2.1.194.1.1`). The top-level OIDs in the subtree under object

ipfixSelectorFunctions MUST be registered in a sub-registry maintained by IANA at <http://www.iana.org/assignments/smi-numbers>. The first entry in this subtree is the Select All function (ipfixFuncSelectAll), defined in this document as {ipfixSelectorFunctions 1}.

New Selector Functions MUST be registered at IANA and are subject to Expert Review [RFC5226], i.e., review by one of a group of experts designated by an IETF Area Director. The group of experts MUST check the requested MIB objects for completeness and accuracy of the description. Requests for MIB objects that duplicate the functionality of existing objects SHOULD be declined. The smallest available OID SHOULD be assigned to new MIB objects. The specification of new MIB objects SHOULD follow the structure specified in Section 6.1 and MUST be published using a well-established and persistent publication medium. The experts will initially be drawn from the Working Group Chairs and document editors of the IPFIX and PSAMP Working Groups.

6.1. The Selector Functions

The following figure shows what the MIB tree usually should look like. It already contains ipfixFuncSelectAll. The subtree in ipfixFuncF2 gives the basic structure that all selection methods SHOULD follow.

```

ipfixSelectorFunctions
|
+- ipfixFuncSelectAll
|   |
|   +- ipfixFuncSelectAllAvail (is the function available?)
|
+- ipfixFuncF2
|   |
|   +- ipfixFuncF2Avail (is the function F2 available?)
|   |
|   +- ipfixFuncF2Parameters (a table with parameters)
|   ...
|
+- ipfixFuncFn...
```

The selection method SHOULD be designed as a MIB subtree introduced by an object with the name ipfixFunc appended by a function name. The objects in this subtree SHOULD be prefixed by this name. If the function is named Fx, then we would start a subtree with an OID named ipfixFuncFx. This subtree should contain an object ipfixFuncFxAvail that has the type TruthValue. If a selection method takes parameters, the MIB should contain a table named

ipfixFuncFxParameters, which should contain all the parameters that the selection method specifies. An entry in this table will be referenced by the IPFIX MIB module if the selection method with the parameters is used.

To illustrate the structure defined above, the following contains an example of a function MyFunc that holds three integer parameters Param1, Param2, and Param3. In the example, there are currently two instances of the parameter sets, defined with indexes 1 and 4.

```
ipfixSelectorFunctions (1)
|
+- ipfixFuncMyFunc (?)
|
+- ipfixFuncMyFuncAvail (1) = true
+- ipfixFuncMyFuncParameters (2)
|
+- ipfixFuncMyFuncParametersEntry (1)
|
|   +- index (1) (ipfixFuncMyFuncParametersIndex)
|   |   +- ipfixFuncMyFuncParam1 (1) = 47
|   |   +- ipfixFuncMyFuncParam2 (2) = -128
|   |   +- ipfixFuncMyFuncParam3 (3) = 19
|   |
|   +- index(4) (ipfixFuncMyFuncParametersIndex)
|   |   +- ipfixFuncMyFuncParam1 (1) = 19
|   |   +- ipfixFuncMyFuncParam2 (2) = -1
|   |   +- ipfixFuncMyFuncParam3 (3) = 728
```

If the function defined above is referenced in the IPFIX MIB module, the ipfixSelectionProcessTable would look as follows:

```
ipfixSelectionProcessTable (8)
|
+- ipfixSelectionProcessEntry (1)
|
|   +- index (9) (ipfixMeteringProcessCacheId)
|   +- index (1) (ipfixSelectionProcessIndex)
|   |   +- index (1) (ipfixSelectionProcessSelectorIndex)
|   |   |   +- ipfixSelectionProcessSelectorFunction (3)
|   |   |   |   = ipfixSelectorFunctions.?.2.1.4
|   |   +- index (2) (ipfixSelectionProcessSelectorIndex)
|   |   |   +- ipfixSelectionProcessSelectorFunction (3)
|   |   |   |   = ipfixSelectorFunctions.?.2.1.1
```

This means that for the `ipfixMeteringProcessCacheId(9)`, a Selection Process with index 1 is created that applies the same function two times but with different parameter sets. First, the function `MyFunc` is applied with the parameters of the set with index 4, and then with the parameters of the set with index 1.

7. Relationship to Other MIB Modules

Besides the usual imports from the SNMP Standards [RFC2578], [RFC2579], and [RFC2580], the IPFIX MIB module references the ENTITY MIB module [RFC4133] and the Interfaces MIB module [RFC2863].

7.1. Relationship to the ENTITY MIB and Interfaces MIB

The Observation Point table (`ipfixObservationPointTable`) contains a reference to the ENTITY MIB module [RFC4133] (`ipfixObservationPointPhysicalEntity`) and a reference to the Interfaces MIB module [RFC2863] (`ipfixObservationPointPhysicalInterface`). If the implementers of the IPFIX MIB module want to specify the physical entity where Flows are observed, then they SHOULD also implement the ENTITY MIB and/or the Interfaces MIB module. The implementation of the ENTITY MIB and/or the Interfaces MIB module is OPTIONAL. If one of them is not implemented, then all values of the respective column `ipfixObservationPointPhysicalEntity` or `ipfixObservationPointPhysicalInterface` in the Observation Point table are zero and the values of the `ipfixObservationPointPhysicalEntityDirection` columns are `unknown(0)`, if none of them are defined.

7.2. MIB Modules Required for IMPORTS

The IPFIX MIB module requires the modules SNMPv2-SMI [RFC2578], SNMPv2-TC [RFC2579], and SNMPv2-CONF [RFC2580]. Further on, it imports the textual conventions `InetAddressType` and `InetAddress` from the INET ADDRESS MIB module [RFC4001].

The IPFIX SELECTOR MIB module also requires the modules SNMPv2-SMI [RFC2578], SNMPv2-TC [RFC2579], and SNMPv2-CONF [RFC2580].

8. MIB Definitions

This section contains the definitions of the IPFIX-MIB module and the IPFIX-SELECTOR-MIB module. There are different mandatory groups defined for Collector and Exporter implementations. The statistical objects are made OPTIONAL.

8.1. IPFIX MIB Definition

```
IPFIX-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
    MODULE-IDENTITY, OBJECT-TYPE, mib-2, Unsigned32, Counter64,
    Gauge32
        FROM SNMPv2-SMI                -- [RFC2578]
    TimeStamp, DateAndTime
        FROM SNMPv2-TC                -- [RFC2579]
    MODULE-COMPLIANCE, OBJECT-GROUP
        FROM SNMPv2-CONF              -- [RFC2580]
    InterfaceIndexOrZero
        FROM IF-MIB                   -- [RFC2863]
    InetAddressType, InetAddress, InetPortNumber
        FROM INET-ADDRESS-MIB         -- [RFC4001]
    PhysicalIndexOrZero
        FROM ENTITY-MIB;              -- [RFC4133]
```

```
ipfixMIB MODULE-IDENTITY
```

```
    LAST-UPDATED "201206110000Z"      -- 11 June 2012
```

```
    ORGANIZATION "IETF IPFIX Working Group"
```

```
    CONTACT-INFO
```

```
        "WG charter:
```

```
        http://www.ietf.org/html.charters/ipfix-charter.html
```

```
    Mailing Lists:
```

```
        General Discussion: ipfix@ietf.org
```

```
        To Subscribe: http://www1.ietf.org/mailman/listinfo/ipfix
```

```
        Archive:
```

```
http://www1.ietf.org/mail-archive/web/ipfix/current/index.html
```

```
    Editor:
```

```
        Thomas Dietz
```

```
        NEC Europe Ltd.
```

```
        NEC Laboratories Europe
```

```
        Network Research Division
```

```
        Kurfuersten-Anlage 36
```

```
        Heidelberg 69115
```

```
        Germany
```

Phone: +49 6221 4342-128
Email: Thomas.Dietz@neclab.eu

Atsushi Kobayashi
NTT Information Sharing Platform Laboratories
3-9-11 Midori-cho
Musashino-shi, Tokyo 180-8585
Japan
Phone: +81-422-59-3978
Email: akoba@nttv6.net

Benoit Claise
Cisco Systems, Inc.
De Kleetlaan 6a b1
Diegem 1831
Belgium
Phone: +32 2 704 5622
Email: bclaise@cisco.com

Gerhard Muenz
Technische Universitaet Muenchen
Department of Informatics
Chair for Network Architectures and Services (I8)
Boltzmannstr. 3
Garching 85748
Germany
Email: muenz@net.in.tum.de"

DESCRIPTION

"The IPFIX MIB defines managed objects for IP Flow Information eXport. These objects provide information about managed nodes supporting the IPFIX protocol, for Exporters as well as for Collectors.

Copyright (c) 2012 IETF Trust and the persons identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, is permitted pursuant to, and subject to the license terms contained in, the Simplified BSD License set forth in Section 4.c of the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>)."

-- Revision history

REVISION	"201206110000Z"	-- 11 June 2012
DESCRIPTION		

"Fixed errata from RFC 5815. Published as RFC 6615."

REVISION "201004190000Z" -- 19 April 2010

DESCRIPTION

"Initial version, published as RFC 5815."

::= { mib-2 193 }

```
--*****
-- Top-Level Structure of the MIB
--*****
```

```
ipfixObjects      OBJECT IDENTIFIER ::= { ipfixMIB 1 }
ipfixConformance OBJECT IDENTIFIER ::= { ipfixMIB 2 }
```

```
ipfixMainObjects OBJECT IDENTIFIER ::= { ipfixObjects 1 }
ipfixStatistics  OBJECT IDENTIFIER ::= { ipfixObjects 2 }
```

```
-----
-- 1.1: Objects Used by All IPFIX Implementations
-----
-----
-- 1.1.1: Transport Session Table
-----
```

```
ipfixTransportSessionTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF IpfixTransportSessionEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table lists the currently established Transport
        Sessions between an Exporting Process and a Collecting
        Process."
    ::= { ipfixMainObjects 1 }
```

```
ipfixTransportSessionEntry OBJECT-TYPE
    SYNTAX      IpfixTransportSessionEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Defines an entry in the ipfixTransportSessionTable."
    INDEX       { ipfixTransportSessionIndex }
    ::= { ipfixTransportSessionTable 1 }
```

```
IpfixTransportSessionEntry ::=
    SEQUENCE {
        ipfixTransportSessionIndex      Unsigned32,
        ipfixTransportSessionProtocol    Unsigned32,
        ipfixTransportSessionSourceAddressType  InetAddressType,
```

```

    ipfixTransportSessionSourceAddress      InetAddress,
    ipfixTransportSessionDestinationAddressType InetAddressType,
    ipfixTransportSessionDestinationAddress InetAddress,
    ipfixTransportSessionSourcePort         InetPortNumber,
    ipfixTransportSessionDestinationPort    InetPortNumber,
    ipfixTransportSessionSctpAssocId        Unsigned32,
    ipfixTransportSessionDeviceMode         INTEGER,
    ipfixTransportSessionTemplateRefreshTimeout Unsigned32,
    ipfixTransportSessionOptionsTemplateRefreshTimeout Unsigned32,
    ipfixTransportSessionTemplateRefreshPacket Unsigned32,
    ipfixTransportSessionOptionsTemplateRefreshPacket Unsigned32,
    ipfixTransportSessionIpfixVersion       Unsigned32,
    ipfixTransportSessionStatus             INTEGER
}

```

ipfixTransportSessionIndex OBJECT-TYPE

SYNTAX Unsigned32 (1..4294967295)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Locally arbitrary, but unique identifier of an entry in the ipfixTransportSessionTable. The value is expected to remain constant from a re-initialization of the entity's network management agent to the next re-initialization."

::= { ipfixTransportSessionEntry 1 }

ipfixTransportSessionProtocol OBJECT-TYPE

SYNTAX Unsigned32 (1..255)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The transport protocol used for receiving or transmitting IPFIX Messages. Protocol numbers are assigned by IANA. A current list of all assignments is available from <<http://www.iana.org/assignments/protocol-numbers/>>."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Section 10."

::= { ipfixTransportSessionEntry 2 }

ipfixTransportSessionSourceAddressType OBJECT-TYPE

SYNTAX InetAddressType { unknown(0), ipv4(1), ipv6 (2) }

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The type of address used for the source address, as specified in RFC 4001. The InetAddressType supported

values are ipv4(1) and ipv6(2). This object is used with protocols (specified in ipfixTransportSessionProtocol) like TCP (6) and UDP (17) that have the notion of addresses. SCTP (132) should use the ipfixTransportSessionSctpAssocId instead. If SCTP (132) or any other protocol without the notion of addresses is used, the object MUST be set to unknown(0)."

::= { ipfixTransportSessionEntry 3 }

ipfixTransportSessionSourceAddress OBJECT-TYPE

SYNTAX InetAddress

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The source address of the Exporter of the IPFIX Transport Session. This value is interpreted according to the value of ipfixTransportSessionAddressType, as specified in RFC 4001. This object is used with protocols (specified in ipfixTransportSessionProtocol) like TCP (6) and UDP (17) that have the notion of addresses. SCTP (132) should use the ipfixTransportSessionSctpAssocId instead. If SCTP (132) or any other protocol without the notion of addresses is used, the object MUST be set to a zero-length string."

::= { ipfixTransportSessionEntry 4 }

ipfixTransportSessionDestinationAddressType OBJECT-TYPE

SYNTAX InetAddressType { unknown(0), ipv4(1), ipv6 (2) }

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The type of address used for the destination address, as specified in RFC 4001. The InetAddressType supported values are ipv4(1) and ipv6(2). This object is used with protocols (specified in ipfixTransportSessionProtocol) like TCP (6) and UDP (17) that have the notion of addresses. SCTP (132) should use the ipfixTransportSessionSctpAssocId instead. If SCTP (132) or any other protocol without the notion of addresses is used, the object MUST be set to unknown(0)."

::= { ipfixTransportSessionEntry 5 }

ipfixTransportSessionDestinationAddress OBJECT-TYPE

SYNTAX InetAddress

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The destination address of the Collector of the IPFIX Transport Session. This value is interpreted according to

the value of `ipfixTransportSessionAddressType`, as specified in RFC 4001. This object is used with protocols (specified in `ipfixTransportSessionProtocol`) like TCP (6) and UDP (17) that have the notion of addresses. SCTP (132) should use the `ipfixTransportSessionSctpAssocId` instead. If SCTP (132) or any other protocol without the notion of addresses is used, the object MUST be set to a zero-length string."

::= { ipfixTransportSessionEntry 6 }

`ipfixTransportSessionSourcePort` OBJECT-TYPE

SYNTAX InetPortNumber

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The transport protocol port number of the Exporter. This object is used with protocols (specified in `ipfixTransportSessionProtocol`) like TCP (6) and UDP (17) that have the notion of ports. SCTP (132) should copy the value of `sctpAssocLocalPort` if the Transport Session is in collecting mode or `sctpAssocRemPort` if the Transport Session is in exporting mode. The association is referenced by the `ipfixTransportSessionSctpAssocId`. If any other protocol without the notion of ports is used, the object MUST be set to zero."

::= { ipfixTransportSessionEntry 7 }

`ipfixTransportSessionDestinationPort` OBJECT-TYPE

SYNTAX InetPortNumber

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The transport protocol port number of the Collector. The default value is 4739 for all currently defined transport protocol types. This object is used with protocols (specified in `ipfixTransportSessionProtocol`) like TCP (6) and UDP (17) that have the notion of ports. SCTP (132) should copy the value of `sctpAssocRemPort` if the Transport Session is in collecting mode or `sctpAssocLocalPort` if the Transport Session is in exporting mode. The association is referenced by the `ipfixTransportSessionSctpAssocId`. If any other protocol without the notion of ports is used, the object MUST be set to zero."

::= { ipfixTransportSessionEntry 8 }

ipfixTransportSessionSctpAssocId OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The association ID used for the SCTP session between the Exporter and the Collector of the IPFIX Transport Session. It is equal to the sctpAssocId entry in the sctpAssocTable defined in the SCTP MIB. This object is only valid if ipfixTransportSessionProtocol has the value 132 (SCTP). In all other cases, the value MUST be zero."

REFERENCE

"RFC 3873, Stream Control Transmission Protocol (SCTP) Management Information Base (MIB)."

::= { ipfixTransportSessionEntry 9 }

ipfixTransportSessionDeviceMode OBJECT-TYPE

SYNTAX INTEGER {
 exporting(1),
 collecting(2)
 }

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The mode of operation of the device for the given Transport Session. This object can have the following values:

exporting(1)

This value MUST be used if the Transport Session is used for exporting Records to other IPFIX Devices; i.e., this device acts as Exporter.

collecting(2)

This value MUST be used if the Transport Session is used for collecting Records from other IPFIX Devices; i.e., this device acts as Collector."

::= { ipfixTransportSessionEntry 10 }

ipfixTransportSessionTemplateRefreshTimeout OBJECT-TYPE

SYNTAX Unsigned32

UNITS "seconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"On Exporters, this object contains the time in seconds after which IPFIX Templates are resent by the Exporter."

On Collectors, this object contains the lifetime in seconds after which a Template becomes invalid when it is not received again within this lifetime.

This object is only valid if ipfixTransportSessionProtocol has the value 17 (UDP). In all other cases, the value MUST be zero."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Sections 10.3.6 and 10.3.7."

::= { ipfixTransportSessionEntry 11 }

ipfixTransportSessionOptionsTemplateRefreshTimeout OBJECT-TYPE

SYNTAX Unsigned32

UNITS "seconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"On Exporters, this object contains the time in seconds after which IPFIX Options Templates are resent by the Exporter.

On Collectors, this object contains the lifetime in seconds after which an Options Template becomes invalid when it is not received again within this lifetime.

This object is only valid if ipfixTransportSessionProtocol has the value 17 (UDP). In all other cases, the value MUST be zero."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Sections 10.3.6 and 10.3.7."

::= { ipfixTransportSessionEntry 12 }

ipfixTransportSessionTemplateRefreshPacket OBJECT-TYPE

SYNTAX Unsigned32

UNITS "packets"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"On Exporters, this object contains the number of exported IPFIX Messages after which IPFIX Templates are resent by the Exporter.

On Collectors, this object contains the lifetime in number of exported IPFIX Messages after which a Template becomes invalid when it is not received again within this lifetime.

This object is only valid if ipfixTransportSessionProtocol has the value 17 (UDP). In all other cases, the value MUST be zero."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Sections 10.3.6 and 10.3.7."

::= { ipfixTransportSessionEntry 13 }

ipfixTransportSessionOptionsTemplateRefreshPacket OBJECT-TYPE

SYNTAX Unsigned32

UNITS "packets"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"On Exporters, this object contains the number of exported IPFIX Messages after which IPFIX Options Templates are resent by the Exporter.

On Collectors, this object contains the lifetime in number of exported IPFIX Messages after which an Options Template becomes invalid when it is not received again within this lifetime.

This object is only valid if ipfixTransportSessionProtocol has the value 17 (UDP). In all other cases, the value MUST be zero."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Sections 10.3.6 and 10.3.7."

::= { ipfixTransportSessionEntry 14 }

ipfixTransportSessionIpfixVersion OBJECT-TYPE

SYNTAX Unsigned32 (0..65535)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"On Exporters, the object contains the version number of the IPFIX protocol that the Exporter uses to export its data in this Transport Session.

On Collectors, the object contains the version number of the IPFIX protocol it receives for this Transport Session.

If IPFIX Messages of different IPFIX protocol versions are transmitted or received in this Transport Session, this object contains the maximum version number."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Section 3.1."

::= { ipfixTransportSessionEntry 15 }

ipfixTransportSessionStatus OBJECT-TYPE

SYNTAX INTEGER {
 unknown(0),
 inactive(1),
 active(2)
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The status of a Transport Session. This object can have the following values:

unknown(0)

This value MUST be used if the status of the Transport Session cannot be detected by the equipment. This value should be avoided as far as possible.

inactive(1)

This value MUST be used for Transport Sessions that are specified in the system but are not currently active. The value can be used, for example, for Transport Sessions that are backup (secondary) sessions in a Transport Session group.

active(2)

This value MUST be used for Transport Sessions that are currently active and transmitting or receiving data."

::= { ipfixTransportSessionEntry 16 }

```
-----
-- 1.1.2: Template Table
-----
```

```
ipfixTemplateTable OBJECT-TYPE
```

```
    SYNTAX      SEQUENCE OF IpfixTemplateEntry
```

```
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "This table lists the Templates and Options Templates that
        are transmitted by the Exporting Process or received by the
        Collecting Process.
```

```
        The table contains the Templates and Options Templates that
        are received or used for exporting data for a given
        Transport Session group and Observation Domain.
```

```
        Withdrawn or invalidated (Options) Templates MUST be removed
        from this table."
```

```
 ::= { ipfixMainObjects 2 }
```

```
ipfixTemplateEntry OBJECT-TYPE
```

```
    SYNTAX      IpfixTemplateEntry
```

```
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "Defines an entry in the ipfixTemplateTable."
```

```
    INDEX      {
        ipfixTransportSessionIndex,
        ipfixTemplateObservationDomainId,
        ipfixTemplateId
    }
```

```
 ::= { ipfixTemplateTable 1 }
```

```
IpfixTemplateEntry ::=
```

```
    SEQUENCE {
```

```
        ipfixTemplateObservationDomainId Unsigned32,
```

```
        ipfixTemplateId Unsigned32,
```

```
        ipfixTemplateSetId Unsigned32,
```

```
        ipfixTemplateAccessTime DateAndTime
```

```
    }
```

```
ipfixTemplateObservationDomainId OBJECT-TYPE
```

```
    SYNTAX      Unsigned32 (0..4294967295)
```

```
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "The ID of the Observation Domain for which this Template
        is defined. This value is used when sending IPFIX Messages.
```

The special value of 0 indicates that the Data Records exported with this (Options Template) cannot be applied to a single Observation Domain."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Section 3.1."

::= { ipfixTemplateEntry 1 }

ipfixTemplateId OBJECT-TYPE

SYNTAX Unsigned32 (256..65535)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This number indicates the Template ID in the IPFIX Message. Values from 0 to 255 are not allowed for Template IDs."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Section 3.4.1."

::= { ipfixTemplateEntry 2 }

ipfixTemplateSetId OBJECT-TYPE

SYNTAX Unsigned32 (1..65535)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This number indicates the Set ID of the Template. This object allows the Template type to be easily retrieved."

Currently, there are two values defined. The value 2 is used for Sets containing Template definitions. The value 3 is used for Sets containing Options Template definitions."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Section 3.3.2."

::= { ipfixTemplateEntry 3 }

ipfixTemplateAccessTime OBJECT-TYPE

SYNTAX DateAndTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"If the Transport Session is in exporting mode (ipfixTransportSessionDeviceMode) the time when this (Options) Template was last sent to the Collector(s)."

In the specific case of UDP as transport protocol, this time is used to know when a retransmission of the (Options) Template is needed.

If the Transport Session is in collecting mode, this object contains the time when this (Options) Template was last received from the Exporter. In the specific case of UDP as transport protocol, this time is used to know when this (Options) Template times out and thus is no longer valid."

```
::= { ipfixTemplateEntry 4 }
```

```
-- 1.1.3: Exported Template Definition Table
```

```
ipfixTemplateDefinitionTable OBJECT-TYPE
```

```
SYNTAX      SEQUENCE OF IpfixTemplateDefinitionEntry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

"On Exporters, this table lists the (Options) Template fields of which a (Options) Template is defined. It defines the (Options) Template given in the ipfixTemplateId specified in the ipfixTemplateTable.

On Collectors, this table lists the (Options) Template fields of which a (Options) Template is defined. It defines the (Options) Template given in the ipfixTemplateId specified in the ipfixTemplateTable."

```
::= { ipfixMainObjects 3 }
```

```
ipfixTemplateDefinitionEntry OBJECT-TYPE
```

```
SYNTAX      IpfixTemplateDefinitionEntry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

"Defines an entry in the ipfixTemplateDefinitionTable."

```
INDEX      {
    ipfixTransportSessionIndex,
    ipfixTemplateObservationDomainId,
    ipfixTemplateId,
    ipfixTemplateDefinitionIndex
}
```

```
::= { ipfixTemplateDefinitionTable 1 }
```

```
IpfixTemplateDefinitionEntry ::=
```

```
SEQUENCE {
    ipfixTemplateDefinitionIndex      Unsigned32,
    ipfixTemplateDefinitionId        Unsigned32,
```



```
    ipfixTemplateDefinitionIeLength      Unsigned32,  
    ipfixTemplateDefinitionEnterpriseNumber Unsigned32,  
    ipfixTemplateDefinitionFlags        BITS  
}
```

ipfixTemplateDefinitionIndex OBJECT-TYPE

```
SYNTAX      Unsigned32 (1..65535)  
MAX-ACCESS  not-accessible  
STATUS      current
```

DESCRIPTION

"The ipfixTemplateDefinitionIndex specifies the order in which the Information Elements are used in the (Options) Template Record.

Since a Template Record can contain a maximum of 65535 Information Elements, the index is limited to this value."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Sections 3.4.1 and 3.4.2."

::= { ipfixTemplateDefinitionEntry 1 }

ipfixTemplateDefinitionIeId OBJECT-TYPE

```
SYNTAX      Unsigned32 (1..65535)  
MAX-ACCESS  read-only  
STATUS      current
```

DESCRIPTION

"This indicates the Information Element ID at position ipfixTemplateDefinitionIndex in the (Options) Template ipfixTemplateId. This implicitly specifies the data type of the Information Element. The elements are registered at IANA. A current list of assignments can be found at <<http://www.iana.org/assignments/ipfix/>>."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Section 3.2.

RFC 5102, Information Model for IP Flow Information Export."

::= { ipfixTemplateDefinitionEntry 2 }

ipfixTemplateDefinitionIeLength OBJECT-TYPE

```
SYNTAX      Unsigned32 (0..65535)  
MAX-ACCESS  read-only  
STATUS      current
```

DESCRIPTION

"This indicates the length of the Information Element ID at position ipfixTemplateDefinitionIndex in the (Options) Template ipfixTemplateId."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Section 3.2.

RFC 5102, Information Model for IP Flow Information Export."

::= { ipfixTemplateDefinitionEntry 3 }

ipfixTemplateDefinitionEnterpriseNumber OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"IANA enterprise number of the authority defining the Information Element identifier in this Template Record. Enterprise numbers are assigned by IANA. A current list of all assignments is available from <<http://www.iana.org/assignments/enterprise-numbers/>>.

This object must be zero(0) for all standard Information Elements registered with IANA. A current list of these elements is available from <<http://www.iana.org/assignments/ipfix/>>."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Section 3.2.

RFC 5102, Information Model for IP Flow Information Export."

::= { ipfixTemplateDefinitionEntry 4 }

ipfixTemplateDefinitionFlags OBJECT-TYPE

SYNTAX BITS {
 scope(0),
 flowKey(1)
}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This bitmask indicates special attributes for the Information Element:

scope(0)

This Information Element is used for scope.

flowKey(1)

This Information Element is a Flow Key.

Thus, we get the following values for an Information Element:

If neither bit scope(0) nor bit flowKey(1) is set

The Information Element is neither used for scoping nor as Flow Key.

If only bit scope(0) is set

The Information Element is used for scoping.

If only bit flowKey(1) is set

The Information Element is used as Flow Key.

Both bit scope(0) and flowKey(1) MUST NOT be set at the same time. This combination is not allowed."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Sections 2 and 3.4.2.1.

RFC 5102, Information Model for IP Flow Information Export."

::= { ipfixTemplateDefinitionEntry 5 }

-- 1.1.4: Export Table

ipfixExportTable OBJECT-TYPE

SYNTAX SEQUENCE OF IpfixExportEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table lists all exports of an IPFIX Device.

On Exporters, this table contains all exports grouped by Transport Session, Observation Domain ID, Template ID, and Metering Process represented by the ipfixMeteringProcessCacheId. Thanks to the ipfixExportIndex, the exports can group one or more Transport Sessions to achieve a special functionality like failover management, load-balancing, etc. The entries with the same ipfixExportIndex, ipfixObservationDomainId, and ipfixMeteringProcessCacheId define a Transport Session group. If the Exporter does not use Transport Session grouping, then each ipfixExportIndex contains a single ipfixMeteringProcessCacheId, and thus a single Transport Session; this session MUST have a member type

value of primary(1). Transport Sessions referenced in this table MUST have a ipfixTransportSessionDeviceMode value of exporting(1).

On Collectors, this table is not needed."

```
::= { ipfixMainObjects 4 }
```

ipfixExportEntry OBJECT-TYPE

SYNTAX IpfixExportEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Defines an entry in the ipfixExportTable."

```
INDEX {
    ipfixExportIndex,
    ipfixMeteringProcessCacheId,
    ipfixTransportSessionIndex
}
```

```
::= { ipfixExportTable 1 }
```

IpfixExportEntry ::=

```
SEQUENCE {
    ipfixExportIndex      Unsigned32,
    ipfixExportMemberType INTEGER
}
```

ipfixExportIndex OBJECT-TYPE

SYNTAX Unsigned32 (1..4294967295)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Locally arbitrary, but unique identifier of an entry in the ipfixExportTable. The value is expected to remain constant from a re-initialization of the entity's network management agent to the next re-initialization."

A common ipfixExportIndex between two entries from this table indicates that there is a relationship between the Transport Sessions in ipfixTransportSessionIndex. The type of relationship is expressed by the value of ipfixExportMemberType."

```
::= { ipfixExportEntry 1 }
```

ipfixExportMemberType OBJECT-TYPE

```
SYNTAX INTEGER {
    unknown(0),
    primary(1),
    secondary(2),
}
```

```
        parallel(3),
        loadBalancing(4)
    }
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
    "The type of member Transport Session in a Transport
    Session group (identified by the value of ipfixExportIndex,
    ipfixObservationDomainId, and ipfixMeteringProcessCacheId).
    The following values are valid:

    unknown(0)
        This value MUST be used if the status of the group
        membership cannot be detected by the equipment.  This
        value should be avoided as far as possible.

    primary(1)
        This value is used for a group member that is used as
        the primary target of an Exporter.  Other group members
        (with the same ipfixExportIndex and
        ipfixMeteringProcessCacheId) MUST NOT have the value
        primary(1) but MUST have the value secondary(2).
        This value MUST also be specified if the Exporter does
        not support Transport Session grouping.  In this case,
        the group contains only one Transport Session.

    secondary(2)
        This value is used for a group member that is used as a
        secondary target of an Exporter.  The Exporter will use
        one of the targets specified as secondary(2) within the
        same Transport Session group when the primary target is
        not reachable.

    parallel(3)
        This value is used for a group member that is used for
        duplicate exporting (i.e., all group members identified
        by the ipfixExportIndex are exporting the same Records
        in parallel).  This implies that all group members MUST
        have the same member type (i.e., parallel(3)).

    loadBalancing(4)
        This value is used for a group member that is used
        as one target for load-balancing.  This means that a
        Record is sent to one of the group members in this
        group identified by ipfixExportIndex.
        This implies that all group members MUST have the same
        member type (i.e., loadBalancing(4))."
 ::= { ipfixExportEntry 2 }
```

-- 1.1.5: Metering Process Table

ipfixMeteringProcessTable OBJECT-TYPE

SYNTAX SEQUENCE OF IpfixMeteringProcessEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table lists so-called caches used at the Metering Process to store the metering data of Flows observed at the Observation Points given in the ipfixObservationPointGroupReference. The table lists the timeouts that specify when the cached metering data is expired.

On Collectors, the table is not needed."

::= { ipfixMainObjects 5 }

ipfixMeteringProcessEntry OBJECT-TYPE

SYNTAX IpfixMeteringProcessEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Defines an entry in the ipfixMeteringProcessTable."

INDEX { ipfixMeteringProcessCacheId }

::= { ipfixMeteringProcessTable 1 }

IpfixMeteringProcessEntry ::=

SEQUENCE {

ipfixMeteringProcessCacheId Unsigned32,

ipfixMeteringProcessObservationPointGroupRef Unsigned32,

ipfixMeteringProcessCacheActiveTimeout Unsigned32,

ipfixMeteringProcessCacheIdleTimeout Unsigned32

}

ipfixMeteringProcessCacheId OBJECT-TYPE

SYNTAX Unsigned32 (1..4294967295)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Locally arbitrary, but unique identifier of an entry in the ipfixMeteringProcessTable. The value is expected to remain constant from a re-initialization of the entity's network management agent to the next re-initialization."

::= { ipfixMeteringProcessEntry 1 }

ipfixMeteringProcessObservationPointGroupRef OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The Observation Point Group ID that links this table entry to the ipfixObservationPointTable. The matching ipfixObservationPointGroupId in that table gives the Observation Points used in that cache. If the Observation Points are unknown, the

ipfixMeteringProcessObservationPointGroupRef MUST be zero."

::= { ipfixMeteringProcessEntry 2 }

ipfixMeteringProcessCacheActiveTimeout OBJECT-TYPE

SYNTAX Unsigned32

UNITS "seconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"On the Exporter, this object contains the time after which a Flow is expired (and a Data Record for the Template is sent), even though packets matching this Flow are still received by the Metering Process. If this value is 0, the Flow is not prematurely expired."

REFERENCE

"RFC 5470, Architecture for IP Flow Information Export, Section 5.1.1, item 3."

::= { ipfixMeteringProcessEntry 3 }

ipfixMeteringProcessCacheIdleTimeout OBJECT-TYPE

SYNTAX Unsigned32

UNITS "seconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"On the Exporter, this object contains the time after which a Flow is expired (and a Data Record for the Template is sent) when no packets matching this Flow are received by the Metering Process for the given number of seconds. If this value is zero, the Flow is expired immediately; i.e., a Data Record is sent for every packet received by the Metering Process."

REFERENCE

"RFC 5470, Architecture for IP Flow Information Export, Section 5.1.1, item 1"

::= { ipfixMeteringProcessEntry 4 }

```
-----
-- 1.1.6: Observation Point Table
-----
```

```
ipfixObservationPointTable OBJECT-TYPE
```

```
    SYNTAX      SEQUENCE OF IpfixObservationPointEntry
```

```
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "This table lists the Observation Points used within an
        Exporter by the Metering Process. The index
        ipfixObservationPointGroupId groups Observation Points
        and is referenced in the Metering Process table.
```

```
        On Collectors, this table is not needed."
```

```
 ::= { ipfixMainObjects 6 }
```

```
ipfixObservationPointEntry OBJECT-TYPE
```

```
    SYNTAX      IpfixObservationPointEntry
```

```
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "Defines an entry in the ipfixObservationPointTable."
```

```
    INDEX      {
        ipfixObservationPointGroupId,
        ipfixObservationPointIndex
    }
```

```
 ::= { ipfixObservationPointTable 1 }
```

```
IpfixObservationPointEntry ::=
```

```
    SEQUENCE {
```

```
        ipfixObservationPointGroupId      Unsigned32,
```

```
        ipfixObservationPointIndex        Unsigned32,
```

```
        ipfixObservationPointObservationDomainId Unsigned32,
```

```
        ipfixObservationPointPhysicalEntity PhysicalIndexOrZero,
```

```
        ipfixObservationPointPhysicalInterface InterfaceIndexOrZero,
```

```
        ipfixObservationPointPhysicalEntityDirection INTEGER
```

```
    }
```

```
ipfixObservationPointGroupId OBJECT-TYPE
```

```
    SYNTAX      Unsigned32 (1..4294967295)
```

```
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "Locally arbitrary, but unique identifier of an entry in the
        ipfixObservationPointTable. The value is expected to remain
        constant from a re-initialization of the entity's network
        management agent to the next re-initialization.
```


This index represents a group of Observation Points.

The special value of 0 MUST NOT be used within this table but is reserved for usage in the ipfixMeteringProcessTable. An index of 0 for the ipfixObservationPointGroupReference index in that table indicates that an Observation Point is unknown or unspecified for a Metering Process cache."

::= { ipfixObservationPointEntry 1 }

ipfixObservationPointIndex OBJECT-TYPE

SYNTAX Unsigned32 (1..4294967295)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Locally arbitrary, but unique identifier of an entry in the ipfixObservationPointTable. The value is expected to remain constant from a re-initialization of the entity's network management agent to the next re-initialization.

This index represents a single Observation Point in an Observation Point group."

::= { ipfixObservationPointEntry 2 }

ipfixObservationPointObservationDomainId OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The ID of the Observation Domain in which this Observation Point is included.

The special value of 0 indicates that the Observation Points within this group cannot be applied to a single Observation Domain."

REFERENCE

"RFC 5101, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, Section 3.1."

::= { ipfixObservationPointEntry 3 }

ipfixObservationPointPhysicalEntity OBJECT-TYPE

SYNTAX PhysicalIndexOrZero

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object contains the index of a physical entity in the ENTITY MIB. This physical entity is the given Observation Point. If such a physical entity cannot be

specified or is not known, then the object is zero."
 ::= { ipfixObservationPointEntry 4 }

ipfixObservationPointPhysicalInterface OBJECT-TYPE

SYNTAX InterfaceIndexOrZero

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object contains the index of a physical interface in the Interfaces MIB. This physical interface is the given Observation Point. If such a physical interface cannot be specified or is not known, then the object is zero.

This object MAY be used alone or in addition to ipfixObservationPointPhysicalEntity. If ipfixObservationPointPhysicalEntity is not zero, this object MUST point to the same physical interface that is referenced in ipfixObservationPointPhysicalEntity. Otherwise, it may reference any interface in the Interfaces MIB."

::= { ipfixObservationPointEntry 5 }

ipfixObservationPointPhysicalEntityDirection OBJECT-TYPE

SYNTAX INTEGER {
 unknown(0),
 ingress(1),
 egress(2),
 both(3)
 }

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The direction of the Flow that is monitored on the given physical entity. The following values are valid:

unknown(0)

This value MUST be used if a direction is not known for the given physical entity.

ingress(1)

This value is used for monitoring incoming Flows on the given physical entity.

egress(2)

This value is used for monitoring outgoing Flows on the given physical entity.

both(3)

This value is used for monitoring incoming and outgoing
Flows on the given physical entity."
::= { ipfixObservationPointEntry 6 }

-- 1.1.7: Selection Process Table

ipfixSelectionProcessTable OBJECT-TYPE

SYNTAX SEQUENCE OF IpfixSelectionProcessEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table contains Selector Functions connected to a
Metering Process by the index ipfixMeteringProcessCacheId.
The Selector Functions are grouped into Selection Processes
by the ipfixSelectionProcessIndex. The Selector Functions
are applied within the Selection Process to the packets
observed for the given Metering Process cache in increasing
order as indicated by the ipfixSelectionProcessSelectorIndex.
This means Selector Functions with a lower
ipfixSelectionProcessSelectorIndex are applied first.
The remaining packets are accounted for in Flow Records.

Since IPFIX does not define any Selector Function (except
selecting every packet), this is a placeholder for future
use and a guideline for implementing enterprise-specific
Selector Function objects.

The following object tree should help the reader visualize
how the Selector Function objects should be implemented:

```
ipfixSelectorFunctions
|
+- ipfixFuncSelectAll
|   |
|   +- ipfixFuncSelectAllAvail (is the function available?)
|
+- ipfixFuncF2
|   |
|   +- ipfixFuncF2Avail (is the function F2 available?)
|   |
|   +- ipfixFuncF2Parameters (a table with parameters)
|   ...
|
+- ipfixFuncFn...
```

If a Selector Function takes parameters, the MIB should contain a table with an entry for each set of parameters used at the Exporter."

::= { ipfixMainObjects 7 }

ipfixSelectionProcessEntry OBJECT-TYPE

SYNTAX IpfixSelectionProcessEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Defines an entry in the ipfixSelectionProcessTable."

INDEX {
ipfixMeteringProcessCacheId,
ipfixSelectionProcessIndex,
ipfixSelectionProcessSelectorIndex
}

::= { ipfixSelectionProcessTable 1 }

IpfixSelectionProcessEntry ::= SEQUENCE {

ipfixSelectionProcessIndex Unsigned32,

ipfixSelectionProcessSelectorIndex Unsigned32,

ipfixSelectionProcessSelectorFunction OBJECT IDENTIFIER

}

ipfixSelectionProcessIndex OBJECT-TYPE

SYNTAX Unsigned32 (1..4294967295)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Locally arbitrary, but unique identifier of an entry in the ipfixSelectionProcessTable. The value is expected to remain constant from a re-initialization of the entity's network management agent to the next re-initialization."

::= { ipfixSelectionProcessEntry 1 }

ipfixSelectionProcessSelectorIndex OBJECT-TYPE

SYNTAX Unsigned32 (1..4294967295)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Index specifying the order in which the referenced ipfixSelectionProcessSelectorFunctions are applied to the observed packet stream within the given Selection Process (identified by the ipfixSelectionProcessIndex). The Selector Functions are applied in increasing order; i.e., Selector Functions with a lower index are applied first."

::= { ipfixSelectionProcessEntry 2 }

ipfixSelectionProcessSelectorFunction OBJECT-TYPE

SYNTAX OBJECT IDENTIFIER

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The pointer to the Selector Function used at position ipfixSelectionProcessSelectorIndex in the list of Selector Functions for the Metering Process cache specified by the index ipfixMeteringProcessCacheId and for the given Selection Process (identified by the ipfixSelectionProcessIndex).

This usually points to an object in the IPFIX SELECTOR MIB. If the Selector Function does not take parameters, then it MUST point to the root of the function subtree. If the function takes parameters, then it MUST point to an entry in the parameter table of the Selector Function."

```
::= { ipfixSelectionProcessEntry 3 }
```

```
-----
-- 1.2.1: Transport Session Statistics Table
-----
```

ipfixTransportSessionStatsTable OBJECT-TYPE

SYNTAX SEQUENCE OF IpfixTransportSessionStatsEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table lists Transport Session statistics between Exporting Processes and Collecting Processes."

```
::= { ipfixStatistics 1 }
```

ipfixTransportSessionStatsEntry OBJECT-TYPE

SYNTAX IpfixTransportSessionStatsEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Defines an entry in the ipfixTransportSessionStatsTable."

AUGMENTS { ipfixTransportSessionEntry }

```
::= { ipfixTransportSessionStatsTable 1 }
```

IpfixTransportSessionStatsEntry ::=

SEQUENCE {

ipfixTransportSessionRate	Gauge32,
ipfixTransportSessionPackets	Counter64,
ipfixTransportSessionBytes	Counter64,
ipfixTransportSessionMessages	Counter64,
ipfixTransportSessionDiscardedMessages	Counter64,
ipfixTransportSessionRecords	Counter64,

```
    ipfixTransportSessionTemplates          Counter64,  
    ipfixTransportSessionOptionsTemplates Counter64,  
    ipfixTransportSessionDiscontinuityTime TimeStamp  
}
```

ipfixTransportSessionRate OBJECT-TYPE

```
SYNTAX      Gauge32  
UNITS       "bytes/second"  
MAX-ACCESS  read-only  
STATUS      current  
DESCRIPTION  
    "The number of bytes per second received by the  
    Collector or transmitted by the Exporter. A  
    value of zero (0) means that no packets were sent or  
    received yet. This object is updated every second."  
::= { ipfixTransportSessionStatsEntry 1 }
```

ipfixTransportSessionPackets OBJECT-TYPE

```
SYNTAX      Counter64  
UNITS       "packets"  
MAX-ACCESS  read-only  
STATUS      current  
DESCRIPTION  
    "The number of packets received by the Collector  
    or transmitted by the Exporter.  
    Discontinuities in the value of this counter can occur at  
    re-initialization of the management system and at other  
    times as indicated by the value of  
    ipfixTransportSessionDiscontinuityTime."  
::= { ipfixTransportSessionStatsEntry 2 }
```

ipfixTransportSessionBytes OBJECT-TYPE

```
SYNTAX      Counter64  
UNITS       "bytes"  
MAX-ACCESS  read-only  
STATUS      current  
DESCRIPTION  
    "The number of bytes received by the Collector  
    or transmitted by the Exporter.  
    Discontinuities in the value of this counter can occur at  
    re-initialization of the management system and at other  
    times as indicated by the value of  
    ipfixTransportSessionDiscontinuityTime."  
::= { ipfixTransportSessionStatsEntry 3 }
```

ipfixTransportSessionMessages OBJECT-TYPE

```
SYNTAX      Counter64  
MAX-ACCESS  read-only
```

STATUS current

DESCRIPTION

"The number of IPFIX Messages received by the Collector or transmitted by the Exporter. Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ipfixTransportSessionDiscontinuityTime."

::= { ipfixTransportSessionStatsEntry 4 }

ipfixTransportSessionDiscardedMessages OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of received IPFIX Messages that are malformed, cannot be decoded, are received in the wrong order, or are missing according to the sequence number.

If used at the Exporter, the number of messages that could not be sent due to, for example, internal buffer overflows, network congestion, or routing issues.

Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ipfixTransportSessionDiscontinuityTime."

::= { ipfixTransportSessionStatsEntry 5 }

ipfixTransportSessionRecords OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of Data Records received by the Collector or transmitted by the Exporter. Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ipfixTransportSessionDiscontinuityTime."

::= { ipfixTransportSessionStatsEntry 6 }

ipfixTransportSessionTemplates OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of Templates received or transmitted. Discontinuities in the value of this counter can occur at

re-initialization of the management system and at other times as indicated by the value of ipfixTransportSessionDiscontinuityTime."
 ::= { ipfixTransportSessionStatsEntry 7 }

ipfixTransportSessionOptionsTemplates OBJECT-TYPE

SYNTAX Counter64
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "The number of Options Templates received or transmitted. Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ipfixTransportSessionDiscontinuityTime."
 ::= { ipfixTransportSessionStatsEntry 8 }

ipfixTransportSessionDiscontinuityTime OBJECT-TYPE

SYNTAX TimeStamp
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "The value of sysUpTime at the most recent occasion at which one or more of the Transport Session counters suffered a discontinuity. A value of zero indicates that no such discontinuity has occurred since the last re-initialization of the local management subsystem."
 ::= { ipfixTransportSessionStatsEntry 9 }

 -- 1.2.2: Template Statistics Table

ipfixTemplateStatsTable OBJECT-TYPE

SYNTAX SEQUENCE OF IpfixTemplateStatsEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "This table lists statistics objects per Template."
 ::= { ipfixStatistics 2 }

ipfixTemplateStatsEntry OBJECT-TYPE

SYNTAX IpfixTemplateStatsEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "Defines an entry in the ipfixTemplateStatsTable."


```

AUGMENTS      { ipfixTemplateEntry }
::= { ipfixTemplateStatsTable 1 }

```

```

IpfixTemplateStatsEntry ::=
    SEQUENCE {
        ipfixTemplateDataRecords      Counter64,
        ipfixTemplateDiscontinuityTime TimeStamp
    }

```

```
ipfixTemplateDataRecords OBJECT-TYPE
```

```

SYNTAX      Counter64
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The number of Data Records that are transmitted or received
    per Template.
    Discontinuities in the value of this counter can occur at
    re-initialization of the management system and at other
    times as indicated by the value of
    ipfixTemplateDiscontinuityTime."
::= { ipfixTemplateStatsEntry 1 }

```

```
ipfixTemplateDiscontinuityTime OBJECT-TYPE
```

```

SYNTAX      TimeStamp
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The value of sysUpTime at the most recent occasion at which
    the Template counter suffered a discontinuity.
    A value of zero indicates that no such discontinuity has
    occurred since the last re-initialization of the local
    management subsystem."
::= { ipfixTemplateStatsEntry 2 }

```

```

-----
-- 1.2.3: Metering Process Statistics Table
-----

```

```
ipfixMeteringProcessStatsTable OBJECT-TYPE
```

```

SYNTAX      SEQUENCE OF IpfixMeteringProcessStatsEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "This table lists statistics objects that have data per
    Metering Process cache.

    On Collectors, this table is not needed."
::= { ipfixStatistics 3 }

```

```
ipfixMeteringProcessStatsEntry OBJECT-TYPE
    SYNTAX      IpfixMeteringProcessStatsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Defines an entry in the ipfixMeteringProcessStatsTable."
    AUGMENTS    { ipfixMeteringProcessEntry }
    ::= { ipfixMeteringProcessStatsTable 1 }

IpfixMeteringProcessStatsEntry ::=
    SEQUENCE {
        ipfixMeteringProcessCacheActiveFlows      Gauge32,
        ipfixMeteringProcessCacheUnusedCacheEntries Gauge32,
        ipfixMeteringProcessCacheDataRecords      Counter64,
        ipfixMeteringProcessCacheDiscontinuityTime TimeStamp
    }

ipfixMeteringProcessCacheActiveFlows OBJECT-TYPE
    SYNTAX      Gauge32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of Flows currently active at this cache."
    ::= { ipfixMeteringProcessStatsEntry 1 }

ipfixMeteringProcessCacheUnusedCacheEntries OBJECT-TYPE
    SYNTAX      Gauge32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of unused cache entries."
    ::= { ipfixMeteringProcessStatsEntry 2 }

ipfixMeteringProcessCacheDataRecords OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of Data Records generated.
        Discontinuities in the value of this counter can occur at
        re-initialization of the management system and at other
        times as indicated by the value of
        ipfixMeteringProcessCacheDiscontinuityTime."
    ::= { ipfixMeteringProcessStatsEntry 3 }

ipfixMeteringProcessCacheDiscontinuityTime OBJECT-TYPE
    SYNTAX      TimeStamp
    MAX-ACCESS  read-only
```

```

STATUS      current
DESCRIPTION
    "The value of sysUpTime at the most recent occasion at which
    the Metering Process counter suffered a discontinuity.
    A value of zero indicates that no such discontinuity has
    occurred since the last re-initialization of the local
    management subsystem."
::= { ipfixMeteringProcessStatsEntry 4 }

```

```
-- 1.2.4: Selection Process Statistics Table
```

```

ipfixSelectionProcessStatsTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF IpfixSelectionProcessStatsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table contains statistics for the Selector Functions
        connected to a Metering Process by the index
        ipfixMeteringProcessCacheId.

        The indexes MUST match an entry in the
        ipfixSelectionProcessTable."
    ::= { ipfixStatistics 4 }

```

```

ipfixSelectionProcessStatsEntry OBJECT-TYPE
    SYNTAX      IpfixSelectionProcessStatsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Defines an entry in the ipfixSelectionProcessStatsTable."
    AUGMENTS    { ipfixSelectionProcessEntry }
    ::= { ipfixSelectionProcessStatsTable 1 }

```

```

IpfixSelectionProcessStatsEntry ::= SEQUENCE {
    ipfixSelectionProcessStatsPacketsObserved    Counter64,
    ipfixSelectionProcessStatsPacketsDropped    Counter64,
    ipfixSelectionProcessStatsDiscontinuityTime TimeStamp
}

```

```

ipfixSelectionProcessStatsPacketsObserved OBJECT-TYPE
    SYNTAX      Counter64
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of packets observed at the entry point of the
        function. The entry point may be the Observation Point or
        the exit point of another Selector Function."

```

Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ipfixSelectionProcessStatsDiscontinuityTime."
 ::= { ipfixSelectionProcessStatsEntry 1 }

ipfixSelectionProcessStatsPacketsDropped OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of packets dropped while selecting packets. Discontinuities in the value of this counter can occur at re-initialization of the management system and at other times as indicated by the value of ipfixSelectionProcessStatsDiscontinuityTime."

::= { ipfixSelectionProcessStatsEntry 2 }

ipfixSelectionProcessStatsDiscontinuityTime OBJECT-TYPE

SYNTAX TimeStamp

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime at the most recent occasion at which one or more of the Selector counters suffered a discontinuity. A value of zero indicates that no such discontinuity has occurred since the last re-initialization of the local management subsystem."

::= { ipfixSelectionProcessStatsEntry 3 }

 -- 2: Conformance Information

 ipfixCompliances OBJECT IDENTIFIER ::= { ipfixConformance 1 }
 ipfixGroups OBJECT IDENTIFIER ::= { ipfixConformance 2 }

 -- 2.1: Compliance Statements

 ipfixCollectorCompliance MODULE-COMPLIANCE

STATUS current

DESCRIPTION

"An implementation that builds an IPFIX Collector that complies with this module MUST implement the objects defined in the mandatory group ipfixCommonGroup."

The implementation of all objects in the other groups is optional and depends on the corresponding functionality implemented in the equipment.

An implementation that is compliant with this MIB module is limited to using only the values TCP (6), UDP (17), and SCTP (132) in the ipfixTransportSessionProtocol object because these are the only protocols currently specified for usage within IPFIX (see RFC 5101)."

```
MODULE -- this module
MANDATORY-GROUPS {
    ipfixCommonGroup
}
```

GROUP ipfixCommonStatsGroup

DESCRIPTION

"These objects should be implemented if the statistics function is implemented in the equipment."

::= { ipfixCompliances 1 }

ipfixExporterCompliance MODULE-COMPLIANCE

STATUS current

DESCRIPTION

"An implementation that builds an IPFIX Exporter that complies with this module MUST implement the objects defined in the mandatory group ipfixCommonGroup. The implementation of all other objects depends on the implementation of the corresponding functionality in the equipment."

```
MODULE -- this module
```

```
MANDATORY-GROUPS {
    ipfixCommonGroup,
    ipfixExporterGroup
}
```

GROUP ipfixCommonStatsGroup

DESCRIPTION

"These objects should be implemented if the statistics function is implemented in the equipment."

GROUP ipfixExporterStatsGroup

DESCRIPTION

"These objects MUST be implemented if statistics functions are implemented in the equipment."

::= { ipfixCompliances 2 }

-- 2.2: MIB Grouping

ipfixCommonGroup OBJECT-GROUP

```
OBJECTS {
    ipfixTransportSessionProtocol,
    ipfixTransportSessionSourceAddressType,
    ipfixTransportSessionSourceAddress,
    ipfixTransportSessionDestinationAddressType,
    ipfixTransportSessionDestinationAddress,
    ipfixTransportSessionSourcePort,
    ipfixTransportSessionDestinationPort,
    ipfixTransportSessionSctpAssocId,
    ipfixTransportSessionDeviceMode,
    ipfixTransportSessionTemplateRefreshTimeout,
    ipfixTransportSessionOptionsTemplateRefreshTimeout,
    ipfixTransportSessionTemplateRefreshPacket,
    ipfixTransportSessionOptionsTemplateRefreshPacket,
    ipfixTransportSessionIpfixVersion,
    ipfixTransportSessionStatus,

    ipfixTemplateSetId,
    ipfixTemplateAccessTime,

    ipfixTemplateDefinitionIeId,
    ipfixTemplateDefinitionIeLength,
    ipfixTemplateDefinitionEnterpriseNumber,
    ipfixTemplateDefinitionFlags
}
STATUS      current
DESCRIPTION
    "The main IPFIX objects."
::= { ipfixGroups 1 }
```

ipfixCommonStatsGroup OBJECT-GROUP

```
OBJECTS {
    ipfixTransportSessionRate,
    ipfixTransportSessionPackets,
    ipfixTransportSessionBytes,
    ipfixTransportSessionMessages,
    ipfixTransportSessionDiscardedMessages,
    ipfixTransportSessionRecords,
    ipfixTransportSessionTemplates,
    ipfixTransportSessionOptionsTemplates,
    ipfixTransportSessionDiscontinuityTime,

    ipfixTemplateDataRecords,
    ipfixTemplateDiscontinuityTime
}
```

```
    }
    STATUS          current
    DESCRIPTION
        "Common statistical objects."
    ::= { ipfixGroups 2 }

ipfixExporterGroup OBJECT-GROUP
    OBJECTS {
        ipfixExportMemberType,

        ipfixMeteringProcessObservationPointGroupRef,
        ipfixMeteringProcessCacheActiveTimeout,
        ipfixMeteringProcessCacheIdleTimeout,

        ipfixObservationPointObservationDomainId,
        ipfixObservationPointPhysicalEntity,
        ipfixObservationPointPhysicalInterface,
        ipfixObservationPointPhysicalEntityDirection,

        ipfixSelectionProcessSelectorFunction
    }
    STATUS          current
    DESCRIPTION
        "The main objects for Exporters."
    ::= { ipfixGroups 3 }

ipfixExporterStatsGroup OBJECT-GROUP
    OBJECTS {
        ipfixMeteringProcessCacheActiveFlows,
        ipfixMeteringProcessCacheUnusedCacheEntries,
        ipfixMeteringProcessCacheDataRecords,
        ipfixMeteringProcessCacheDiscontinuityTime,

        ipfixSelectionProcessStatsPacketsObserved,
        ipfixSelectionProcessStatsPacketsDropped,
        ipfixSelectionProcessStatsDiscontinuityTime
    }
    STATUS          current
    DESCRIPTION
        "The statistical objects for Exporters."
    ::= { ipfixGroups 4 }

END
```

8.2. IPFIX SELECTOR MIB Definition

```
IPFIX-SELECTOR-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, mib-2
        FROM SNMPv2-SMI                    -- [RFC2578]
    TruthValue
        FROM SNMPv2-TC                    -- [RFC2579]
    MODULE-COMPLIANCE, OBJECT-GROUP
        FROM SNMPv2-CONF;                -- [RFC2580]

ipfixSelectorMIB MODULE-IDENTITY
    LAST-UPDATED "201206110000Z"           -- 11 June 2012
    ORGANIZATION "IETF IPFIX Working Group"
    CONTACT-INFO
        "WG charter:
         http://www.ietf.org/html.charters/ipfix-charter.html

        Mailing Lists:
         General Discussion: ipfix@ietf.org
         To Subscribe: http://www1.ietf.org/mailman/listinfo/ipfix
         Archive:
         http://www1.ietf.org/mail-archive/web/ipfix/current/index.html

        Editor:
         Thomas Dietz
         NEC Europe Ltd.
         NEC Laboratories Europe
         Network Research Division
         Kurfuersten-Anlage 36
         Heidelberg 69115
         Germany
         Phone: +49 6221 4342-128
         Email: Thomas.Dietz@neclab.eu

         Atsushi Kobayashi
         NTT Information Sharing Platform Laboratories
         3-9-11 Midori-cho
         Musashino-shi, Tokyo 180-8585
         Japan
         Phone: +81-422-59-3978
         Email: akoba@nttv6.net

         Benoit Claise
         Cisco Systems, Inc.
         De Kleetlaan 6a b1
         Diegem 1831
```


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

Gerhard Muenz
Technische Universitaet Muenchen
Department of Informatics
Chair for Network Architectures and Services (I8)
Boltzmannstr. 3
Garching 85748
Germany
Email: muenz@net.in.tum.de"

DESCRIPTION

"The IPFIX SELECTOR MIB module defined in this section provides the standard Filtering and Sampling functions that can be referenced in the ipfixSelectionProcessTable. All standard Filtering and Sampling functions MUST be registered in the subtree under object ipfixSelectorFunctions (1.3.6.1.2.1.194.1.1). The top-level OIDs in the subtree under object ipfixSelectorFunctions MUST be registered in a sub-registry maintained by IANA at <<http://www.iana.org/assignments/smi-numbers/>>.

New Selector Functions MUST be registered at IANA and are subject to Expert Review [RFC5226], i.e., review by one of a group of experts designated by an IETF Area Director. The group of experts MUST check the requested MIB objects for completeness and accuracy of the description. Requests for MIB objects that duplicate the functionality of existing objects SHOULD be declined. The smallest available OID SHOULD be assigned to new MIB objects. The specification of new MIB objects SHOULD follow the structure specified in [RFC6615] and MUST be published using a well-established and persistent publication medium. The experts will initially be drawn from the Working Group Chairs and document editors of the IPFIX and PSAMP Working Groups.

Copyright (c) 2012 IETF Trust and the persons identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, is permitted pursuant to, and subject to the license terms contained in, the Simplified BSD License set forth in Section 4.c of the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>)."

-- Revision history

REVISION "201206110000Z" -- 11 June 2012
 DESCRIPTION
 "Update to MIB description to reflect updated registration
 of new Sampling and Filtering functions. Published as
 RFC 6615."

REVISION "201003150000Z" -- 15 March 2010
 DESCRIPTION
 "Initial version, published as RFC 5815."

::= { mib-2 194 }

--*****
 -- Top-Level Structure of the MIB
 --*****

ipfixSelectorObjects OBJECT IDENTIFIER
 ::= { ipfixSelectorMIB 1 }
 ipfixSelectorConformance OBJECT IDENTIFIER
 ::= { ipfixSelectorMIB 2 }

 -- 1: Objects Used by All IPFIX Implementations

 -- 1.1: Packet Selector Functions for IPFIX

ipfixSelectorFunctions OBJECT IDENTIFIER
 ::= { ipfixSelectorObjects 1 }

 -- 1.1.1: Function 1: Selecting All Packets

ipfixFuncSelectAll OBJECT IDENTIFIER
 ::= { ipfixSelectorFunctions 1 }

ipfixFuncSelectAllAvail OBJECT-TYPE
 SYNTAX TruthValue
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "This object indicates the availability of the trivial
 function of selecting all packets. This function is always
 available."
 ::= { ipfixFuncSelectAll 1 }

```
-----
-- 2: Conformance Information
-----
ipfixSelectorCompliances OBJECT IDENTIFIER
    ::= { ipfixSelectorConformance 1 }
ipfixSelectorGroups      OBJECT IDENTIFIER
    ::= { ipfixSelectorConformance 2 }

-----
-- 2.1: Compliance Statements
-----
ipfixSelectorBasicCompliance MODULE-COMPLIANCE
    STATUS      current
    DESCRIPTION
        "An implementation that builds an IPFIX Exporter that
        complies with this module MUST implement the objects defined
        in the mandatory group ipfixBasicGroup. The implementation
        of all other objects depends on the implementation of the
        corresponding functionality in the equipment."
    MODULE -- this module
    MANDATORY-GROUPS {
        ipfixSelectorBasicGroup
    }
    ::= { ipfixSelectorCompliances 1 }

-----
-- 2.2: MIB Grouping
-----
ipfixSelectorBasicGroup OBJECT-GROUP
    OBJECTS {
        ipfixFuncSelectAllAvail
    }
    STATUS      current
    DESCRIPTION
        "The main IPFIX objects."
    ::= { ipfixSelectorGroups 1 }

END
```

9. Security Considerations

There are no management objects defined in this MIB module that have a MAX-ACCESS clause of read-write and/or read-create. So, if this MIB module is implemented correctly, then there is no risk that an intruder can alter or create any management objects of this MIB module via direct SNMP SET operations.

Some of the readable objects in this MIB module (i.e., objects with a MAX-ACCESS other than not-accessible) may be considered sensitive or vulnerable in some network environments. It is thus important to control even GET and/or NOTIFY access to these objects and possibly to even encrypt the values of these objects when sending them over the network via SNMP. These are the tables and objects and their sensitivity/vulnerability:

- o ipfixTransportSessionTable - contains configuration data that might be sensitive because objects in this table may reveal information about the network infrastructure
- o ipfixExportTable - contains configuration data that might be sensitive because objects in this table may reveal information about the network infrastructure as well
- o ipfixMeteringProcessTable - contains configuration data that might be sensitive because objects in this table may reveal information about the IPFIX Device itself
- o ipfixObservationPointTable - contains configuration data that might be sensitive because objects in this table may reveal information about the IPFIX Device itself and the network infrastructure
- o ipfixSelectorFunctions - currently contains no sensitive data but might want to be secured anyway, since it may contain sensitive data in a future version

All other objects and tables contain no data that is considered sensitive.

SNMP versions prior to SNMPv3 did not include adequate security. Even if the network itself is secure (for example by using IPsec), there is no control as to who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in this MIB module.

Implementations MUST provide the security features described by the SNMPv3 framework (see [RFC3410]), including full support for authentication and privacy via the User-based Security Model (USM) [RFC3414] with the AES cipher algorithm [RFC3826]. Implementations MAY also provide support for the Transport Security Model (TSM) [RFC5591] in combination with a secure transport such as SSH [RFC5592] or TLS/DTLS [RFC6353].

Further, deployment of SNMP versions prior to SNMPv3 is NOT RECOMMENDED. Instead, it is RECOMMENDED to deploy SNMPv3 and to enable cryptographic security. It is then a customer/operator responsibility to ensure that the SNMP entity giving access to an instance of this MIB module is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

10. IANA Considerations

The MIB module in this document uses the following IANA-assigned OBJECT IDENTIFIER values recorded in the SMI Numbers registry:

Descriptor	OBJECT IDENTIFIER value
-----	-----
ipfixMIB	{ mib-2 193 }
ipfixSelectorMIB	{ mib-2 194 }

The IPFIX SELECTOR MIB registry as defined in [RFC5815] Section 10 has been removed by IANA, as its use is discontinued with this document.

IANA has created and maintains a sub-registry at <http://www.iana.org/assignments/smi-numbers>, in which the top-level OIDs in the subtree under object ipfixSelectorFunctions MUST be registered. The initial version of this sub-registry should contain the following:

Sub-registry Name: IPFIX-SELECTOR-MIB Functions
 Reference: [RFC6615]
 Registration Procedures: Expert Review [RFC5226]

Prefix: iso.org.dod.internet.mgmt.
 mib-2.ipfixSelectorMIB.ipfixSelectorObjects.ipfixSelectorFunctions
 (1.3.6.1.2.1.194.1.1)

Decimal	Name	Description	Reference
-----	-----	-----	-----
1	ipfixFuncSelectAll	Select everything	[RFC6615]

Additions to this sub-registry are subject to Expert Review [RFC5226], i.e., review by one of a group of experts designated by an IETF Area Director. The group of experts MUST check the requested MIB objects for completeness and accuracy of the description. Requests for MIB objects that duplicate the functionality of existing objects SHOULD be declined. The smallest available OID SHOULD be assigned to new MIB objects. The specification of new MIB objects SHOULD follow the structure specified in Section 6.1 and MUST be published using a well-established and persistent publication medium. The experts will initially be drawn from the Working Group Chairs and document editors of the IPFIX and PSAMP Working Groups.

11. Acknowledgments

This document is a product of the IPFIX Working Group. The authors would like to thank the following persons: Paul Aitken for his detailed review, Dan Romascanu and the MIB doctors, and many more, for their technical reviews and feedback.

12. References

12.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2578] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.
- [RFC2579] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.
- [RFC2580] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.
- [RFC2863] McCloghrie, K. and F. Kastenholtz, "The Interfaces Group MIB", RFC 2863, June 2000.
- [RFC3873] Pastor, J. and M. Belinchon, "Stream Control Transmission Protocol (SCTP) Management Information Base (MIB)", RFC 3873, September 2004.
- [RFC4001] Daniele, M., Haberman, B., Routhier, S., and J. Schoenwaelder, "Textual Conventions for Internet Network Addresses", RFC 4001, February 2005.

- [RFC4133] Bierman, A. and K. McCloghrie, "Entity MIB (Version 3)", RFC 4133, August 2005.
- [RFC5101] Claise, B., Ed., "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information", RFC 5101, January 2008.
- [RFC5102] Quittek, J., Bryant, S., Claise, B., Aitken, P., and J. Meyer, "Information Model for IP Flow Information Export", RFC 5102, January 2008.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.
- [RFC5815] Dietz, T., Ed., Kobayashi, A., Claise, B., and G. Muenz, "Definitions of Managed Objects for IP Flow Information Export", RFC 5815, April 2010.

12.2. Informative References

- [RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.
- [RFC3414] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", STD 62, RFC 3414, December 2002.
- [RFC3826] Blumenthal, U., Maino, F., and K. McCloghrie, "The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model", RFC 3826, June 2004.
- [RFC3917] Quittek, J., Zseby, T., Claise, B., and S. Zander, "Requirements for IP Flow Information Export (IPFIX)", RFC 3917, October 2004.
- [RFC5470] Sadasivan, G., Brownlee, N., Claise, B., and J. Quittek, "Architecture for IP Flow Information Export", RFC 5470, March 2009.
- [RFC5472] Zseby, T., Boschi, E., Brownlee, N., and B. Claise, "IP Flow Information Export (IPFIX) Applicability", RFC 5472, March 2009.
- [RFC5474] Duffield, N., Ed., Chiou, D., Claise, B., Greenberg, A., Grossglauser, M., and J. Rexford, "A Framework for Packet Selection and Reporting", RFC 5474, March 2009.

- [RFC5475] Zseby, T., Molina, M., Duffield, N., Niccolini, S., and F. Raspall, "Sampling and Filtering Techniques for IP Packet Selection", RFC 5475, March 2009.
- [RFC5476] Claise, B., Ed., Johnson, A., and J. Quittek, "Packet Sampling (PSAMP) Protocol Specifications", RFC 5476, March 2009.
- [RFC5591] Harrington, D. and W. Hardaker, "Transport Security Model for the Simple Network Management Protocol (SNMP)", RFC 5591, June 2009.
- [RFC5592] Harrington, D., Salowey, J., and W. Hardaker, "Secure Shell Transport Model for the Simple Network Management Protocol (SNMP)", RFC 5592, June 2009.
- [RFC6353] Hardaker, W., "Transport Layer Security (TLS) Transport Model for the Simple Network Management Protocol (SNMP)", RFC 6353, July 2011.

Authors' Addresses

Thomas Dietz (editor)
NEC Europe Ltd.
NEC Laboratories Europe
Network Research Division
Kurfuersten-Anlage 36
Heidelberg 69115
DE

Phone: +49 6221 4342-128
EMail: Thomas.Dietz@neclab.eu

Atsushi Kobayashi
NTT Information Sharing Platform Laboratories
3-9-11 Midori-cho
Musashino-shi, Tokyo 180-8585
JA

Phone: +81-422-59-3978
EMail: akoba@nttv6.net

Benoit Claise
Cisco Systems, Inc.
De Kleetlaan 6a b1
Diegem 1831
BE

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

Gerhard Muenz
Technische Universitaet Muenchen
Department of Informatics
Chair for Network Architectures and Services (I8)
Boltzmannstr. 3
Garching 85748
DE

EMail: muenz@net.in.tum.de

