

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 12, 2011

T. Dietz, Ed.
NEC Europe Ltd.
B. Claise
Cisco Systems, Inc.
J. Quittek
NEC Europe Ltd.
November 8, 2010

Definitions of Managed Objects for Packet Sampling
<draft-ietf-ipfix-psamp-mib-02.txt>

Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it describes extensions to the IPFIX MIB module [RFC5815]. For IPFIX implementations that use packet sampling (PSAMP) techniques as described in [RFC5475], this memo defines the PSAMP MIB module containing managed objects for providing information on applied packet selection functions and their parameters.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on May 12, 2011.

Copyright Notice

Copyright (c) 2010 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. Open Issues/TODOs	3
2. The Internet-Standard Management Framework	3
3. Introduction	3
4. PSAMP Documents Overview	4
5. Related IPFIX Documents	4
6. Structure of the PSAMP MIB module	4
6.1. Textual Conventions	5
6.2. Packet Selection Functions	6
6.2.1. Systematic Count-based Sampling	6
6.2.2. Systematic Time-based Sampling	6
6.2.3. Random n-out-of-N Sampling	7
6.2.4. Uniform Probabilistic Sampling	7
6.2.5. Property Match Filtering	7
6.2.6. Hash-based Filtering	8
7. Definitions	8
8. Security Considerations	24
9. IANA Considerations	25
10. Acknowledgment	26
11. References	26
11.1. Normative References	26
11.2. Informative References	26
Authors' Addresses	27

1. Open Issues/TODOs

- o security considerations: check security issues raised by Nick Duffield on privacy issues with hash parameters etc.

2. 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 is compliant to the SMIV2, which is described in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580 [RFC2580].

3. Introduction

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

This document is a product of the IP Flow Information eXport (IPFIX) working group. Work on this document was started in the Packet Sampling (PSAMP) Working Group (WG) and moved to the IPFIX WG when the PSAMP WG was concluded.

Its purpose is to define managed objects for monitoring PSAMP Devices performing packet selection by sampling and hashing as described in [RFC5475].

It is assumed that packet sampling is performed according to the framework defined in [RFC5474].

Managed objects in the PSAMP MIB module are defined as an extension of the IPFIX MIB module [RFC5815]. Since the IPFIX MIB module is for monitoring only the same holds true for the PSAMP MIB module defined in this document. The definition of objects is in line with the PSAMP information model [RFC5477].

Section 6 describes the structure of the PSAMP MIB module and section 7 contains the formal definition. Security issues are discussed in

section 8.

4. PSAMP Documents Overview

[RFC5474]: "A Framework for Packet Selection and Reporting" describes the PSAMP framework for network elements to select subsets of packets by statistical and other methods, and to export a stream of reports on the selected packets to a Collector.

[RFC5475]: "Sampling and Filtering Techniques for IP Packet Selection" describes the set of packet selection techniques supported by PSAMP.

[RFC5476]: "Packet Sampling (PSAMP) Protocol Specifications" specifies the export of packet information from a PSAMP Exporting Process to a PSAMP Collecting Process.

[RFC5477]: "Information Model for Packet Sampling Exports" defines an information and data model for PSAMP.

This document: "Definitions of Managed Objects for Packet Sampling" describes the PSAMP Management Information Base.

5. Related IPFIX Documents

The IPFIX protocol provides network administrators with access to IP Flow information. 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. This document also specifies the data types used in this MIB module and their encoding. The IPFIX MIB [RFC5815] is the basis for this document and is extended by this MIB module.

6. Structure of the PSAMP MIB module

The IPFIX MIB module defined in [RFC5815] has the concept of a packet selection process containing a set of selector function instances. Selection processes and functions are referenced in the `ipfixSelectionProcessTable` of the IPFIX MIB module. This table identifies an instance of a selector function by an OID. The OID points to an object that describes the selector function. For simple selector functions without parameters, the OID refers to an object that only contains one more object indicating the current availability of this function. For functions that have one or more parameters the object has a subtree that in addition to an

availability object contains a table with a conceptual column for each parameter. Entries (conceptual rows) in this table represent different combinations of parameter values for instances of the selector function.

Object `ipfixSelectorFunctions` in the IPFIX SELECTOR MIB module serves as home for objects that describe instances of packet selector functions. The IPFIX SELECTOR MIB is a very small module that is also defined in [RFC5815]. Objects under `ipfixSelectorFunctions` are maintained by IANA. In the IPFIX SELECTOR module object `ipfixSelectorFunctions` contains just a single trivial packet selector function called `ipfixFuncSelectAll` that selects every packet and has no parameter:

```
ipfixSelectorMIB
+- ipfixSelectorObjects(1)
  +- ipfixSelectorFunctions(1)
    +- ipfixFuncSelectAll(1)
      +- ipfixFuncSelectAllAvail(1)
```

The PSAMP MIB module defined in this document contains six new objects under `ipfixSelectorFunctions`. Each of them describes a packet selector function with one or more parameters. Naming and ordering of objects is fully in line with the guidelines given in section 6.1 of [RFC5815]. All functions and their parameters are already listed in the overview of functions given by the figure in section 8.2.1 of [RFC5477].

In addition, the PSAMP MIB module contains two textual conventions that define data types used for parameters in the above tables that cannot be expressed by the basic data types defined by [RFC2578], `unsigned64` and `float64`.

6.1. Textual Conventions

The MIB module defines one textual conventions that defines a data type used within this MIB module. Another data type is imported from the APPLICATION MIB [RFC2564]. Those data types are defined according to [RFC5101]. Those data types are not integral part of [RFC2578] but are needed to define objects in this MIB module that conform to the Information Elements defined for those objects in [RFC5477].

The `Unsigned64TC` textual convention describes an unsigned integer of 64 bits. It is imported from the APPLICATION MIB.

The `PsampFloat64` textual convention describes a double precision floating point number. It is encoded according to [IEEE.754.1985].

6.2. Packet Selection Functions

In general, different packet selector functions have different parameters. The PSAMP MIB module contains six objects with subtrees that provide information on parameters of function instances of different selector functions. All objects are named and structured according to section 8.2.1 of [RFC5477]:

```
ipfixSelectorFunctions(1)
+-- psampSampCountBased(2)
+- -psampSampTimeBased(3)
+-- psampSampRandOutOfN(4)
+-- psampSampUniProb(5)
+-- psampFiltPropMatch(6)
+-- psampFiltHash(7)
```

Indexing of these functions in the PSAMP MIB module starts with index (2). The function ipfixFuncSelectAll with index (1) is already defined in the IPFIX SELECTOR MIB module.

The object tree for each of these functions is described below. Semantics of all functions and their parameters are described in detail in [RFC5475]. More information on the Selector Reports can also be found in section 6.5.2 of [RFC5476].

6.2.1. Systematic Count-based Sampling

The first selector function is systematic count-based sampling. Its availability is indicated by object psampSampCountBasedAvail. The function has two parameters: psampSampCountBasedInterval and psampSampCountBasedSpace. Different combination of values of these parameters for different instances of the selector function are represented by different conceptual rows in table psampSampCountBasedParamSetEntry:

```
psampSampCountBased(2)
+-- psampSampCountBasedAvail(1)
+-- psampSampCountBasedParamSetTable(2)
    +-- psampSampCountBasedParamSetEntry(1) [psampSampCountBasedIndex]
        +-- psampSampCountBasedIndex(1)
        +-- psampSampCountBasedInterval(2)
        +-- psampSampCountBasedSpace(3)
```

6.2.2. Systematic Time-based Sampling

The second selector function is systematic time-based sampling. The structure of the sub-tree for this function is similar to the previous one. Parameters are psampSampTimeBasedInterval and

psampSampTimeBasedSpace. They appear to be the same as for count based sampling, but their data types are different because they indicate time values instead of numbers of packets:

```
psampSampTimeBased(3)
+-- psampSampTimeBasedAvail(1)
+-- psampSampTimeBasedParamSetTable(2)
    +-- psampSampTimeBasedParamSetEntry(1) [psampSampTimeBasedIndex]
        +-- psampSampTimeBasedIndex(1)
        +-- psampSampTimeBasedInterval(2)
        +-- psampSampTimeBasedSpace(3)
```

6.2.3. Random n-out-of-N Sampling

The third selector function is random n-out-of-N sampling. The structure of the sub-tree for this function is similar to the previous one. Parameters are psampSampRandOutOfNSamplingSize and psampSampRandOutOfNPopulation:

```
psampSampRandOutOfN(4)
+-- psampSampRandOutOfNAvail(1)
+-- psampSampRandOutOfNParamSetTable(3)
    +-- psampSampRandOutOfNParamSetEntry(1) [psampSampRandOutOfNIndex]
        +-- psampSampRandOutOfNIndex(1)
        +-- psampSampRandOutOfNSamplingSize(2)
        +-- psampSampRandOutOfNPopulation(3)
```

6.2.4. Uniform Probabilistic Sampling

The fourth selector function is uniform probabilistic sampling. It has just a single parameter called psampSampUniProbProbability:

```
psampSampUniProb(5)
+-- psampSampUniProbAvail(1)
+-- psampSampUniProbParamSetTable(3)
    +-- psampSampUniProbParamSetEntry(1) [psampSampUniProbIndex]
        +-- psampSampUniProbIndex(1)
        +-- psampSampUniProbProbability(2)
```

6.2.5. Property Match Filtering

The fifth selector function is property match filtering. For this selector function there is a broad variety of possible parameters that could be used. But as stated in section 8.2.1 of [RFC5477] there are no agreed parameters specified and the sub-tree for this function only contains an object indicating the availability of this function. Parameters cannot be retrieved via the PSAMP MIB module:

```
psampFiltPropMatch(6)
+-- psampFiltPropMatchAvail(1)
```

6.2.6. Hash-based Filtering

The sixth selector function is hash-based filtering. This function has more parameters and the actual number may vary with the choice of the hash function applied. The common parameter set for all hash-based filtering functions contains 7 parameters:

psampFiltHashInitializerValue, psampFiltHashIpPayloadOffset, psampFiltHashIpPayloadSize, psampFiltHashSelectedRangeMin, psampFiltHashSelectedRangeMax, psampFiltHashOutputRangeMin, and psampFiltHashOutputRangeMax.

```
psampFiltHash(7)
+-- psampFiltHashAvail(1)
+-- psampFiltHashCapabilities(2)
+-- psampFiltHashParamSetTable(3)
+-- psampFiltHashParamSetEntry(1) [psampFiltHashIndex]
+-- psampFiltHashIndex(1)
+-- psampFiltHashFunction(2)
+-- psampFiltHashInitializerValue(3)
+-- psampFiltHashIpPayloadOffset(4)
+-- psampFiltHashIpPayloadSize(5)
+-- psampFiltHashSelectedRangeMin(6)
+-- psampFiltHashSelectedRangeMax(7)
+-- psampFiltHashOutputRangeMin(8)
+-- psampFiltHashOutputRangeMax(9)
```

Further parameters depend on the applied hash function and are not specified within the PSAMP MIB module.

7. Definitions

PSAMP-MIB DEFINITIONS ::= BEGIN

IMPORTS

```
MODULE-IDENTITY, OBJECT-TYPE, Integer32, Unsigned32, mib-2
    FROM SNMPv2-SMI                -- RFC2578
TEXTUAL-CONVENTION, TruthValue
    FROM SNMPv2-TC                -- RFC2579
MODULE-COMPLIANCE, OBJECT-GROUP
    FROM SNMPv2-CONF              -- RFC2580
Unsigned64TC
    FROM APPLICATION-MIB          -- RFC2564
ipfixSelectorFunctions
```


FROM IPFIX-SELECTOR-MIB;

psampMIB MODULE-IDENTITY

LAST-UPDATED "201011081200Z" -- 08 November 2010

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

69115 Heidelberg

Germany

Phone: +49 6221 4342-128

Email: Thomas.Dietz@nw.neclab.eu

Benoit Claise

Cisco Systems, Inc.

De Kleetlaan 6a b1

Degem 1831

Belgium

Phone: +32 2 704 5622

Email: bclaise@cisco.com

Juergen Quittek

NEC Europe Ltd.

NEC Laboratories Europe

Network Research Division

Kurfuersten-Anlage 36

69115 Heidelberg

Germany

Phone: +49 6221 4342-115

Email: quittek@nw.neclab.eu

DESCRIPTION

"The PSAMP MIB defines managed objects for packet sampling and filtering.

These objects provide information about managed nodes supporting packet sampling, including packet sampling

```
capabilities, configuration and statistics.

Copyright (c) 2010 IETF Trust and the persons identified as
the document authors. All rights reserved. This version
of this MIB module is part of RFC yyyy; see the RFC itself
for full legal notices"
-- RFC Ed.: replace yyyy with actual RFC number & remove this notice

-- Revision history

REVISION      "201011081200Z"          -- 08 November 2010
DESCRIPTION
    "Initial version, published as RFC yyyy."
-- RFC Ed.: replace yyyy with actual RFC number & remove this notice

 ::= { mib-2 xxx }
-- xxx to be assigned by IANA.

PsampFloat64 ::= TEXTUAL-CONVENTION
    STATUS      current
    DESCRIPTION
        "Represents the float64 data type and MUST be encoded as
        an IEEE double-precision 64-bit floating point-type, as
        specified in IEEE 754."
    SYNTAX      OCTET STRING (SIZE (8))

-- Top level structure of the MIB

psampObjects      OBJECT IDENTIFIER ::= { psampMIB 1 }
psampConformance OBJECT IDENTIFIER ::= { psampMIB 2 }

=====
-- Packet selection sampling methods group of objects
=====

=====
--* Method 1: Systematic count-based Sampling
=====

-- Reference: RFC5475, Section 5.1, RFC5476 Section 6.5.2.1 and
--             RFC5477, Section 8.2
psampSampCountBased OBJECT IDENTIFIER
    ::= { ipfixSelectorFunctions 2 }

psampSampCountBasedAvail OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS   read-only
```

```

STATUS      current
DESCRIPTION
    "This object indicates the availability of systematic
    count-based sampling at the managed node.

    A Selector may be unavailable if it is implemented but
    currently disabled due to e.g., administrative reasons, lack
    of resources or similar."
DEFVAL { false }
 ::= { psampSampCountBased 1 }

-- Parameter Set Table ++++++

psampSampCountBasedParamSetTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF
                  PsampSampCountBasedParamSetEntry
    MAX-ACCESS   not-accessible
    STATUS      current
    DESCRIPTION
        "This table lists configurations of systematic count-based
        packet sampling. A parameter set describing a
        configuration contains two parameters: the sampling
        interval length and the space."
    ::= { psampSampCountBased 2 }

psampSampCountBasedParamSetEntry OBJECT-TYPE
    SYNTAX      PsampSampCountBasedParamSetEntry
    MAX-ACCESS   not-accessible
    STATUS      current
    DESCRIPTION
        "Defines an entry in the psampSampCountBasedParamSetTable."
    INDEX { psampSampCountBasedIndex }
    ::= { psampSampCountBasedParamSetTable 1 }

PsampSampCountBasedParamSetEntry ::=
    SEQUENCE {
        psampSampCountBasedIndex      Integer32,
        psampSampCountBasedInterval   Unsigned32,
        psampSampCountBasedSpace      Unsigned32
    }

psampSampCountBasedIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..2147483647)
    MAX-ACCESS   not-accessible
    STATUS      current
    DESCRIPTION
        "The index of this parameter set in the
        psampSampCountBasedParamSetTable. It is used in the

```

```
        object ipfixSelectionProcessSelectorFunctionentries of
            the ipfixSelectionProcessTable in the IPFIX-MIB as reference
            to this parameter set."
 ::= { psampSampCountBasedParamSetEntry 1 }

psampSampCountBasedInterval OBJECT-TYPE
    SYNTAX      Unsigned32
    UNITS       "packets"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "This object specifies the number of packets that are
        consecutively sampled. A value of 100 means that 100
        consecutive packets are sampled."
    REFERENCE
        "RFC5475, Section 5.1 and RFC5477, Section 8.2"
 ::= { psampSampCountBasedParamSetEntry 2 }

psampSampCountBasedSpace OBJECT-TYPE
    SYNTAX      Unsigned32
    UNITS       "packets"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "This object specifies the number of packets between two
        psampSampCountBasedInterval's. A value of 100 means that
        the next interval starts 100 packets (which are not sampled)
        after the current psampSampCountBasedInterval is over."
    REFERENCE
        "RFC5475, Section 5.1 and RFC5477, Section 8.2"
 ::= { psampSampCountBasedParamSetEntry 3 }

=====
--* Method 2: Systematic time-based Sampling
=====

-- Reference: RFC5475, Section 5.1, RFC5476 Section 6.5.2.2 and
--             RFC5477, Section 8.2
psampSampTimeBased OBJECT IDENTIFIER
 ::= { ipfixSelectorFunctions 3 }

psampSampTimeBasedAvail OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "This object indicates the availability of systematic
        time-based sampling at the managed node.
```

A Selector may be unavailable if it is implemented but currently disabled due to e.g., administrative reasons, lack of resources or similar."

```

DEFVAL { false }
 ::= { psampSampTimeBased 1 }

-- Parameter Set Table ++++++
psampSampTimeBasedParamSetTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF
                  PsampSampTimeBasedParamSetEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "This table lists configurations of systematic time-based
        packet sampling. A parameter set describing a configuration
        contains two parameters: the sampling interval length and
        the space."
    ::= { psampSampTimeBased 2 }

psampSampTimeBasedParamSetEntry OBJECT-TYPE
    SYNTAX      PsampSampTimeBasedParamSetEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "Defines an entry in the psampSampTimeBasedParamSetTable."
    INDEX { psampSampTimeBasedIndex }
    ::= { psampSampTimeBasedParamSetTable 1 }

PsampSampTimeBasedParamSetEntry ::=
    SEQUENCE {
        psampSampTimeBasedIndex      Integer32,
        psampSampTimeBasedInterval    Unsigned32,
        psampSampTimeBasedSpace       Unsigned32
    }

psampSampTimeBasedIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..2147483647)
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "The index of this parameter set in the
        psampSampTimeBasedParamSetTable. It is used in the
        object ipfixSelectionProcessSelectorFunctionentries of
        the ipfixSelectionProcessTable in the IPFIX-MIB as reference
        to this parameter set."
    ::= { psampSampTimeBasedParamSetEntry 1 }

```

psampSampTimeBasedInterval OBJECT-TYPE

SYNTAX Unsigned32

UNITS "microseconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object specifies the time interval in microseconds during which all arriving packets are sampled."

REFERENCE

"RFC5475, Section 5.1 and RFC5477, Section 8.2"

::= { psampSampTimeBasedParamSetEntry 2 }

psampSampTimeBasedSpace OBJECT-TYPE

SYNTAX Unsigned32

UNITS "microseconds"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object specifies the time interval in microseconds between two psampSampTimeBasedInterval's. A value of 100 means that the next interval starts 100 microseconds (during which no packets are sampled) after the current psampSampTimeBasedInterval is over."

REFERENCE

"RFC5475, Section 5.1 and RFC5477, Section 8.2"

::= { psampSampTimeBasedParamSetEntry 3 }

--* Method 3: Random n-out-of-N Sampling

-- Reference: RFC5475, Section 5.2.1, RFC5476 Section 6.5.2.3 and
-- RFC5477, Section 8.2

psampSampRandOutOfN OBJECT IDENTIFIER

::= { ipfixSelectorFunctions 4 }

psampSampRandOutOfNAvail OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object indicates the availability of random n-out-of-N sampling at the managed node."

A Selector may be unavailable if it is implemented but currently disabled due to e.g., administrative reasons, lack of resources or similar."

DEFVAL { false }

```

 ::= { psampSampRandOutOfN 1 }

-- Parameter Set Table ++++++

psampSampRandOutOfNParamSetTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF
                  PsampSampRandOutOfNParamSetEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table lists configurations of random n-out-of-N
        sampling.  A parameter set describing a configuration
        contains a two parameter only, the sampling size and the
        parent population."
    ::= { psampSampRandOutOfN 3 }

psampSampRandOutOfNParamSetEntry OBJECT-TYPE
    SYNTAX      PsampSampRandOutOfNParamSetEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Defines an entry in the psampSampRandOutOfNParamSetTable."
    INDEX { psampSampRandOutOfNIndex }
    ::= { psampSampRandOutOfNParamSetTable 1 }

PsampSampRandOutOfNParamSetEntry ::=
    SEQUENCE {
        psampSampRandOutOfNIndex      Integer32,
        psampSampRandOutOfNSamplingSize Unsigned32,
        psampSampRandOutOfNPopulation Unsigned32
    }

psampSampRandOutOfNIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..2147483647)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The index of this parameter set in the
        psampSampRandOutOfNParamSetTable.  It is used in the
        object ipfixSelectionProcessSelectorFunctionentries of
        the ipfixSelectionProcessTable in the IPFIX-MIB as reference
        to this parameter set."
    ::= { psampSampRandOutOfNParamSetEntry 1 }

psampSampRandOutOfNSamplingSize OBJECT-TYPE
    SYNTAX      Unsigned32
    UNITS       "packets"
    MAX-ACCESS  read-only

```

```
STATUS      current
DESCRIPTION
    "This object specifies the number of elements taken from the
    parent Population for specified in
    psampSampRandOutOfNPopulation."
REFERENCE
    "RFC5475, Section 5.2.1 and RFC5477, Section 8.2"
 ::= { psampSampRandOutOfNParamSetEntry 2 }

psampSampRandOutOfNPopulation OBJECT-TYPE
    SYNTAX      Unsigned32
    UNITS        "packets"
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "This object specifies the number of elements in the parent
        Population."
    REFERENCE
        "RFC5475, Section 5.2.1 and RFC5477, Section 8.2"
    ::= { psampSampRandOutOfNParamSetEntry 3 }

=====
--* Method 4: Uniform probabilistic Sampling
=====

-- Reference: RFC5475, Section 5.2.2, RFC5476 Section 6.5.2.4 and
--              RFC5477, Section 8.2
psampSampUniProb OBJECT IDENTIFIER ::= { ipfixSelectorFunctions 5 }

psampSampUniProbAvail OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS   read-only
    STATUS      current
    DESCRIPTION
        "This object indicates the availability of random uniform
        probabilistic sampling at the managed node.

        A Selector may be unavailable if it is implemented but
        currently disabled due to e.g., administrative reasons, lack
        of resources or similar."
    DEFVAL { false }
    ::= { psampSampUniProb 1 }

-- Parameter Set Table ++++++

-- Reference: RFC5475, Section 5.2.2.1 and RFC5477, Section 8.2
psampSampUniProbParamSetTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF
```



```

        PsampSampUniProbParamSetEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "This table lists configurations of random probabilistic
    sampling. A parameter set describing a configuration
    contains a single parameter only: the sampling probability."
 ::= { psampSampUniProb 3 }

psampSampUniProbParamSetEntry OBJECT-TYPE
SYNTAX PsampSampUniProbParamSetEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "Defines an entry in the psampSampUniProbParamSetTable."
INDEX { psampSampUniProbIndex }
 ::= { psampSampUniProbParamSetTable 1 }

PsampSampUniProbParamSetEntry ::=
SEQUENCE {
    psampSampUniProbIndex Integer32,
    psampSampUniProbProbability PsampFloat64
}

psampSampUniProbIndex OBJECT-TYPE
SYNTAX Integer32 (1..2147483647)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
    "The index of this parameter set in the
    psampSampUniProbParamSetTable. It is used in the
    object ipfixSelectionProcessSelectorFunctionentries of
    the ipfixSelectionProcessTable in the IPFIX-MIB as reference
    to this parameter set."
 ::= { psampSampUniProbParamSetEntry 1 }

psampSampUniProbProbability OBJECT-TYPE
SYNTAX PsampFloat64
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "This object specifies the probability that a packet is
    sampled, expressed as a value between 0 and 1. The
    probability is equal for every packet. A value of 0 means
    no packet was sampled since the probability is 0. A value
    of 1 means all packets were sampled since the
    probability is 1."
REFERENCE
```

```

    "RFC5475, Section 5.2.2.1 and RFC5477, Section 8.2"
    ::= { psampSampUniProbParamSetEntry 2 }

=====
-- Packet selection filtering methods group of objects
=====

--* Method 5: Property Match filtering
=====

-- Reserves Method 5 (see RFC5475, Section 6.1, RFC5476
-- Section 6.5.2.5 and RFC5477)
psampFiltPropMatch OBJECT IDENTIFIER
    ::= { ipfixSelectorFunctions 6 }

psampFiltPropMatchAvail OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "This object indicates the availability of property match
        filtering at the managed node.

        A Selector may be unavailable if it is implemented but
        currently disabled due to e.g., administrative reasons, lack
        of resources or similar."
    DEFVAL { false }
    ::= { psampFiltPropMatch 1 }

=====
--* Method 1: Hash filtering
=====

-- Reference: RFC5475, Section 6.2, RFC5476 Section 6.5.2.6 and
-- RFC5477, Section 8.3
psampFiltHash OBJECT IDENTIFIER ::= { ipfixSelectorFunctions 7 }

psampFiltHashAvail OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "This object indicates the availability of hash filtering
        at the managed node.

        A Selector may be unavailable if it is implemented but
        currently disabled due to e.g., administrative reasons, lack
```

```
        of resources or similar."
    DEFVAL { false }
    ::= { psampFiltHash 1 }

psampFiltHashCapabilities OBJECT IDENTIFIER
    ::= { psampFiltHash 2 }

-- Parameter Set Table ++++++

-- Reference: RFC5475, Sections 6.2, 3.8, and 7.1
psampFiltHashParamSetTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF
                    PsampFiltHashParamSetEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This table lists configurations of hash filtering. A
        parameter set describing a configuration contains eight
        parameter describing the hash function."
    ::= { psampFiltHash 3 }

psampFiltHashParamSetEntry OBJECT-TYPE
    SYNTAX          PsampFiltHashParamSetEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "Defines an entry in the psampFiltHashParamSetTable."
    INDEX { psampFiltHashIndex }
    ::= { psampFiltHashParamSetTable 1 }

PsampFiltHashParamSetEntry ::=
    SEQUENCE {
        psampFiltHashIndex          Integer32,
        psampFiltHashFunction       INTEGER,
        psampFiltHashInitializerValue Unsigned64TC,
        psampFiltHashIpPayloadOffset Unsigned64TC,
        psampFiltHashIpPayloadSize  Unsigned64TC,
        psampFiltHashSelectedRangeMin Unsigned64TC,
        psampFiltHashSelectedRangeMax Unsigned64TC,
        psampFiltHashOutputRangeMin Unsigned64TC,
        psampFiltHashOutputRangeMax Unsigned64TC
    }

psampFiltHashIndex OBJECT-TYPE
    SYNTAX          Integer32 (1..2147483647)
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
```

```

    "The index of this parameter set in the
      psampFiltHashParamSetTable. It is used in the
      object ipfixSelectionProcessSelectorFunctionentries of
      the ipfixSelectionProcessTable in the IPFIX-MIB as reference
      to this parameter set."
 ::= { psampFiltHashParamSetEntry 1 }

psampFiltHashFunction OBJECT-TYPE
    SYNTAX      INTEGER {
                        crc32(1),
                        ipsx(2),
                        bob(3)
                    }
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "The Hash Function used by this filter. The PSAMP-MIB
        supports the following Hash Functions:

        crc32(1)

        ipsx(2)

        bob(3)
        "
 ::= { psampFiltHashParamSetEntry 2 }

psampFiltHashInitializerValue OBJECT-TYPE
    SYNTAX      Unsigned64TC
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "This object specifies the initializer value to the hash
        function."
    REFERENCE
        "RFC5475, Sections 6.2, 3.8, and 7.1"
 ::= { psampFiltHashParamSetEntry 3 }

psampFiltHashIpPayloadOffset OBJECT-TYPE
    SYNTAX      Unsigned64TC
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "This object specifies the IP payload offset used by a
        Hash-based Selection Selector."
    REFERENCE
        "RFC5475, Sections 6.2, 3.8, and 7.1"
 ::= { psampFiltHashParamSetEntry 4 }
```

psampFiltHashIpPayloadSize OBJECT-TYPE
SYNTAX Unsigned64TC
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "This object specifies the IP payload size used by a
 Hash-based Selection Selector."
REFERENCE
 "RFC5475, Sections 6.2, 3.8, and 7.1"
 ::= { psampFiltHashParamSetEntry 5 }

psampFiltHashSelectedRangeMin OBJECT-TYPE
SYNTAX Unsigned64TC
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "This object specifies the value for the beginning of a hash
 function's selected range."
REFERENCE
 "RFC5475, Sections 6.2, 3.8, and 7.1"
 ::= { psampFiltHashParamSetEntry 6 }

psampFiltHashSelectedRangeMax OBJECT-TYPE
SYNTAX Unsigned64TC
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "This object specifies the value for the end of a hash
 function's selected range."
REFERENCE
 "RFC5475, Sections 6.2, 3.8, and 7.1"
 ::= { psampFiltHashParamSetEntry 7 }

psampFiltHashOutputRangeMin OBJECT-TYPE
SYNTAX Unsigned64TC
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "This object specifies the value for the beginning of a hash
 function's potential output range."
REFERENCE
 "RFC5475, Sections 6.2, 3.8, and 7.1"
 ::= { psampFiltHashParamSetEntry 8 }

psampFiltHashOutputRangeMax OBJECT-TYPE
SYNTAX Unsigned64TC
MAX-ACCESS read-only
STATUS current

```
DESCRIPTION
    "This object specifies the value for the end of a hash
    function's potential output range."
REFERENCE
    "RFC5475, Sections 6.2, 3.8, and 7.1"
    ::= { psampFiltHashParamSetEntry 9 }

=====
-- Conformance information
=====

psampCompliances OBJECT IDENTIFIER ::= { psampConformance 1 }
psampGroups      OBJECT IDENTIFIER ::= { psampConformance 2 }

=====
-- Compliance statements
=====

psampCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        "The implementation of all objects is optional and depends
        on the implementation of the corresponding functionality in
        the equipment."
    MODULE -- this module
        GROUP psampGroupSampCountBased
        DESCRIPTION
            "These objects must be implemented if the corresponding
            sampling function is implemented in the equipment."
        GROUP psampGroupSampTimeBased
        DESCRIPTION
            "These objects must be implemented if the corresponding
            sampling function is implemented in the equipment."
        GROUP psampGroupSampRandOutOfN
        DESCRIPTION
            "These objects must be implemented if the corresponding
            sampling function is implemented in the equipment."
        GROUP psampGroupSampUniProb
        DESCRIPTION
            "These objects must be implemented if the corresponding
            sampling function is implemented in the equipment."
        GROUP psampGroupFiltPropMatch
        DESCRIPTION
            "These objects must be implemented if the corresponding
            filter function is implemented in the equipment."
        GROUP psampGroupFiltHash
        DESCRIPTION
            "These objects must be implemented if the corresponding
```

```
        filter function is implemented in the equipment."
 ::= { psampCompliances 1 }

=====
-- MIB groupings
=====

psampGroupSampCountBased OBJECT-GROUP
  OBJECTS {
    psampSampCountBasedAvail,
    psampSampCountBasedInterval,
    psampSampCountBasedSpace
  }
  STATUS current
  DESCRIPTION
    "These objects are needed if count based sampling is
    implemented."
  ::= { psampGroups 2 }

psampGroupSampTimeBased OBJECT-GROUP
  OBJECTS {
    psampSampTimeBasedAvail,
    psampSampTimeBasedInterval,
    psampSampTimeBasedSpace
  }
  STATUS current
  DESCRIPTION
    "These objects are needed if time based sampling is
    implemented."
  ::= { psampGroups 3 }

psampGroupSampRandOutOfN OBJECT-GROUP
  OBJECTS {
    psampSampRandOutOfNAvail,
    psampSampRandOutOfNSamplingSize,
    psampSampRandOutOfNPopulation
  }
  STATUS current
  DESCRIPTION
    "These objects are needed if random n-out-of-N sampling is
    implemented."
  ::= { psampGroups 4 }

psampGroupSampUniProb OBJECT-GROUP
  OBJECTS {
    psampSampUniProbAvail,
    psampSampUniProbProbability
  }
```

```
STATUS current
DESCRIPTION
    "These objects are needed if uniform probabilistic sampling
    is implemented."
 ::= { psampGroups 5 }

psampGroupFiltPropMatch OBJECT-GROUP
    OBJECTS {
        psampFiltPropMatchAvail
    }
    STATUS current
    DESCRIPTION
        "These objects are needed if property match filtering is
        implemented."
    ::= { psampGroups 6 }

psampGroupFiltHash OBJECT-GROUP
    OBJECTS {
        psampFiltHashAvail,
        psampFiltHashFunction,
        psampFiltHashInitializerValue,
        psampFiltHashIpPayloadOffset,
        psampFiltHashIpPayloadSize,
        psampFiltHashSelectedRangeMin,
        psampFiltHashSelectedRangeMax,
        psampFiltHashOutputRangeMin,
        psampFiltHashOutputRangeMax
    }
    STATUS current
    DESCRIPTION
        "These objects are needed if hash filtering is implemented."
    ::= { psampGroups 9 }

END
```

8. 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 All tables - they contain configuration data that might be sensitive because objects in this table may reveal information about the network infrastructure and device configuration

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.

It is RECOMMENDED that implementers consider the security features as provided by the SNMPv3 framework (see [RFC3410], section 8), including full support for the SNMPv3 cryptographic mechanisms (for authentication and privacy).

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.

9. 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 -----
psampMIB	{ mib-2 xxx }
psampSampCountBased	{ ipfixSelectorFunctions 2 }
psampSampTimeBased	{ ipfixSelectorFunctions 3 }
psampSampRandOutOfN	{ ipfixSelectorFunctions 4 }
psampSampUniProb	{ ipfixSelectorFunctions 5 }
psampFiltPropMatch	{ ipfixSelectorFunctions 6 }
psampFiltHash	{ ipfixSelectorFunctions 7 }

Other than that this document does not impose any IANA considerations.

10. Acknowledgment

This document is a product of the PSAMP and IPFIX working groups.

11. References

11.1. 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., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIV2", STD 58, RFC 2580, April 1999.
- [RFC2564] Kalbfleisch, C., Krupczak, C., Presuhn, R., and J. Saperia, "Application Management MIB", RFC 2564, May 1999.
- [RFC5101] Claise, B., "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information", RFC 5101, January 2008.
- [RFC5477] Dietz, T., Claise, B., Aitken, P., Dressler, F., and G. Carle, "Information Model for Packet Sampling Exports", RFC 5477, March 2009.
- [RFC5815] Dietz, T., Kobayashi, A., Claise, B., and G. Muenz, "Definitions of Managed Objects for IP Flow Information Export", RFC 5815, April 2010.
- [IEEE.754.1985] Institute of Electrical and Electronics Engineers, "Standard for Binary Floating-Point Arithmetic", IEEE Standard 754, August 1985.

11.2. 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.

- [RFC5474] Duffield, N., 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., Johnson, A., and J. Quittek, "Packet Sampling (PSAMP) Protocol Specifications", RFC 5476, March 2009.

Authors' Addresses

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

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

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

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

Juergen Quittek
NEC Europe Ltd.
NEC Laboratories Europe
Kurfuersten-Anlage 36
Heidelberg 69115
DE

Phone: +49 6221 4342-115
Email: quittek@neclab.eu

