`

Internet Draft
Document: draft-ietf-psamp-framework-13.txt
Intended status: Informational
Expires: December 2008

Nick Duffield (Editor)
AT&T Labs - Research

June 27, 2008

A Framework for Packet Selection and Reporting

Status of this Memo

Copyright Notice

Abstract

This document specifies a framework for the PSAMP (Packet SAMPling) protocol. The functions of this protocol are to select packets from a stream according to a set of standardized selectors, to form a stream of reports on the selected packets,

and to export the reports to a collector. This framework details the components of this architecture, then describes some generic

requirements, motivated by the dual aims of ubiquitous deployment and utility of the reports for applications. Detailed requirements for selection, reporting and exporting are described, along with configuration requirements of the PSAMP functions.

Table of Contents

1. Introduction

This document 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.

The motivation for the PSAMP standard comes from the need for measurement-based support for network management and control across multivendor domains. This requires domain-wide consistency in the types of selection schemes available, and the manner in which the resulting measurements are presented and interpreted.

The motivation for specific packet selection operations comes from the applications that they enable. Development of the PSAMP standard is open to influence by the requirements of standards in related IETF Working Groups, for example, IP Performance Metrics

(IPPM) [RFC-2330] and Internet Traffic Engineering (TEWG).

The name PSAMP is a contraction of the phrase Packet Sampling. The word "sampling" captures the idea that only a subset of all

packets passing a network element will be selected for reporting. But PSAMP selection operations include random selection, deterministic selection (filtering), and deterministic approximations to random selection (hash-based selection).

2. PSAMP Documents Overview

PSAMP-FW: "A Framework for Packet Selection and Reporting" (this document). This document 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. Definitions of terminology and the use of the terms "must", "should" and "may" in this document are informational only.

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

[PSAMP-PROTO]: "Packet Sampling (PSAMP) Protocol Specifications" specifies the export of packet information from a PSAMP Exporting Process to a PSAMP Colleting Process

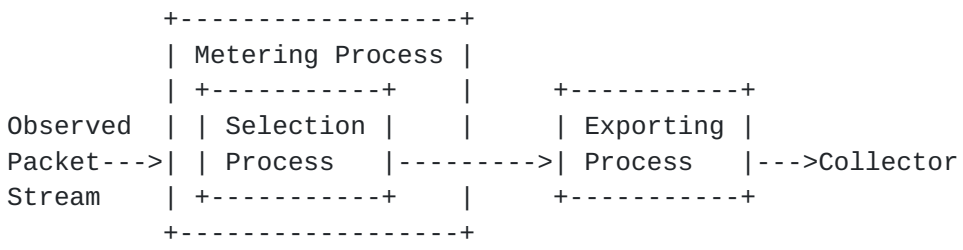
[PSAMP-INFO]: "Information Model for Packet Sampling Exports" defines an information and data model for PSAMP.

3. Elements, Terminology and High-level Architecture

3.1 High-level description of the PSAMP Architecture

Here is an informal high level description of the PSAMP protocol operating in a PSAMP Device (all terms will be defined presently). A stream of packets is observed at an Observation Point. A Selection Process inspects each packet to determine whether or not it is to be selected from reporting. The Selection Process is part of the Metering Process, which constructs a report on each selected packet, using the Packet Content, and possibly other information such as the packet treatment at the Observation Point or the arrival timestamp. An Exporting Process sends the Packet Reports to a Collector, together with any subsidiary information needed for their interpretation.

The following figure indicates the sequence of the three processes (Selection, Metering, and Exporting) within the PSAMP device.



The following sections give the detailed definitions of each of all the objects just named.

3.2 Observation Points, Packet Streams and Packet Content

This section contains the definition of terms relevant to obtaining the packet input to the selection process.

* Observation Point

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

  Note that every Observation Point is associated with an Observation Domain (defined below), and that one Observation Point may be a superset of several other Observation Points. For example one Observation Point can be an entire line card. That would be the superset of the individual Observation Points at the line card's interfaces.

* Observed Packet Stream

  The Observed Packet Stream is the set of all packets observed at the Observation Point.

* Packet Stream

  A Packet Stream denotes a subset of the Observed Packet Stream that flows past some specified point within the Selection Process.
  An example of a Packet Stream is the output of the Selection Process. Note that packets selected from a stream, e.g. by sampling, do not necessarily possess a property by which they can be distinguished from packets that have not been selected. For this reason the term "stream" is favored over "flow", which is defined as set of packets with common properties [RFC-3917].

* Packet Content

The Packet Content denotes the union of the packet header (which includes link layer, network layer and other encapsulation headers) and the packet payload.

3.3 Selection Process

This section defines the selection process and related objects.

* Selection Process

A Selection Process takes the Observed Packet Stream as its input and selects a subset of that stream as its output.

* Selection State:

A Selection Process may maintain state information for use by the Selection Process. At a given time, the Selection State may depend on packets observed at and before that time, and other variables. Examples include:

(i) sequence numbers of packets at the input of Selectors;

(ii) a timestamp of observation of the packet at the Observation Point;

(iii) iterators for pseudorandom number generators;

(iv) hash values calculated during selection;

(v) indicators of whether the packet was selected by a given Selector.

Selection Processes may change portions of the Selection State as a result of processing a packet. Selection state for a packet is to reflect the state after processing the packet.

* Selector:

A Selector defines the action of a Selection Process on a single packet of its input. If selected, the packet becomes an element of the output Packet Stream.

The Selector can make use of the following information in determining whether a packet is selected:

(i) the Packet Content;

(ii) information derived from the packet's treatment at the Observation Point;

(iii) any selection state that may be maintained by the Selection Process.

* Composite Selector:

  A Composite Selector is an ordered composition of Selectors, in which the output Packet Stream issuing from one Selector forms the input Packet Stream to the succeeding Selector.

* Primitive Selector:

  A Selector is primitive if it is not a Composite Selector.

3.4 Reporting

* Packet Reports

  Packet Reports comprise a configurable subset of a packet's input to the Selection Process, including the Packet Content, information relating to its treatment (for example, the output interface), and its associated selection state (for example, a hash of the Packet Content).

* Report Interpretation:

  Report Interpretation comprises subsidiary information, relating to one or more packets, that are used for interpretation of their Packet Reports. Examples include configuration parameters of the Selection Process.

* Report Stream:

  The Report Stream is the output of a Metering Process, comprising two distinguished types of information: Packet Reports, and Report Interpretation.

3.5 Metering Process

A Metering Process selects packets from the Observed Packet Stream using a Selection Process, and produces as output a Report Stream concerning the selected packets.

The PSAMP Metering Process can be viewed as analogous to the IPFIX metering process [RFC-5101], which produces flow records as

its output. While the Metering Process definition in this document specifies the PSAMP definition, the PSAMP protocol specifications [PSAMP-PROTO] will use the IPFIX Metering Process definition, which also suits the PSAMP requirements. The relationship between PSAMP and IPFIX is described more in [PSAMP-INFO] and [PSAMP-PROTO].

3.6 Exporting Process

* Exporting Process:

  An Exporting Process sends, in the form of Export Packets, the output of one or more Metering Processes to one or more Collectors.

* Export Packets:

  An Export Packet is a combination of Report Interpretation(s) and/or one or more Packet Reports that are bundled by the Exporting Process into a Export Packet for exporting to a Collector.

3.7 PSAMP Device

A PSAMP Device is a device hosting at least an Observation Point, a Metering Process (which includes a Selection Process) and an Exporting Process. Typically, corresponding Observation Point(s), Metering Process(es) and Exporting Process(es) are co-located at this device, for example at a router.

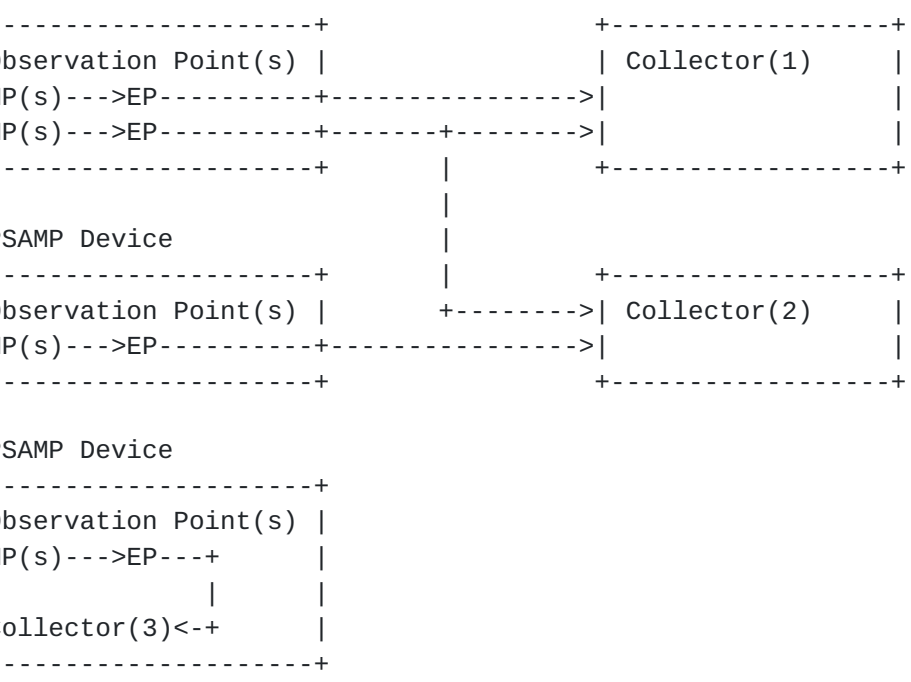
3.8 Collector

A Collector receives a Report Stream exported by one or more Exporting Processes. In some cases, the host of the Metering and/or Exporting Processes may also serve as the Collector.

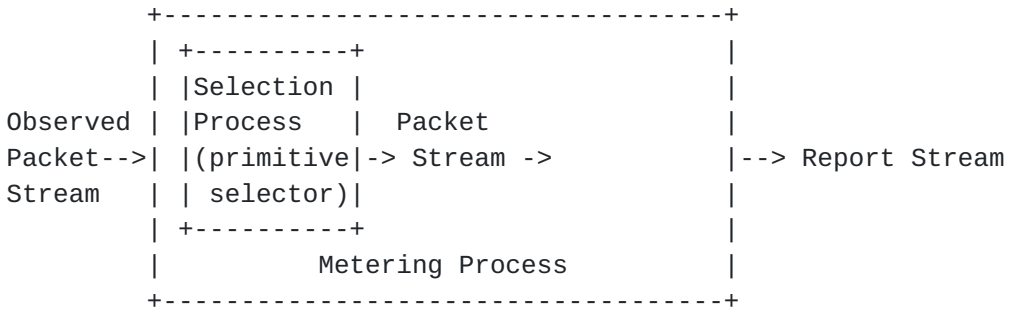
3.9 Possible Configurations

Various possibilities for the high level architecture of these elements are as follows.

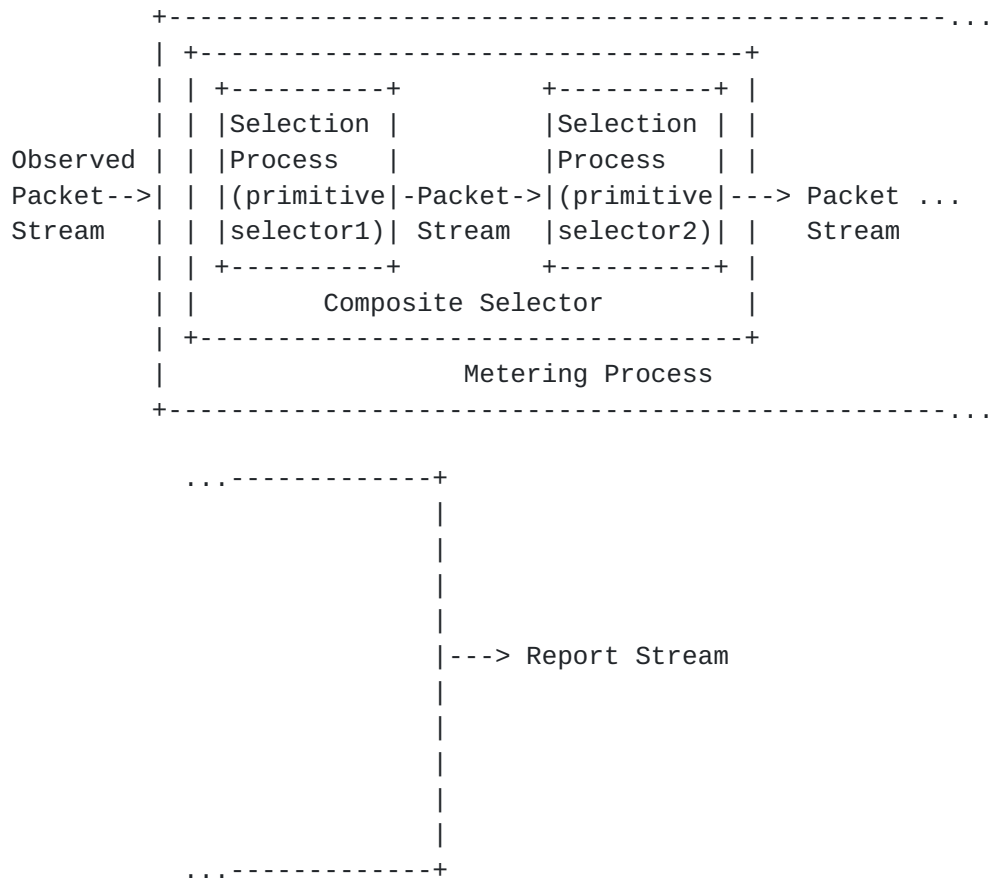
MP = Metering Process, EP = Exporting process



The most simple Metering Process configuration is composed of:



A Metering Process with a composite selector is composed of:



4. Generic Requirements for PSAMP

This section describes the generic requirements for the PSAMP protocol. A number of these are realized as specific requirements in later sections.

4.1 Generic Selection Process Requirements.

(a) Ubiquity: The Selectors must be simple enough to be implemented ubiquitously at maximal line rate.

(b) Applicability: the set of Selectors must be rich enough to support a range of existing and emerging measurement based applications and protocols. This requires a workable trade-off between the range of traffic engineering applications and operational tasks it enables, and the complexity of the set of capabilities.

(c) Extensibility: the protocol must be able to accommodate additional packet Selectors not currently defined.

(d) Flexibility: the protocol must support selection of packets using various network protocols or encapsulation layers,

including Internet Protocol Version 4 (IPv4) [RFC-0791], Internet Protocol Version 6 (IPv6) [RFC-2460], and Multiprotocol Label Switching (MPLS) [RFC-3031].

(e) Robust Selection: packet selection must be robust against attempts to craft an Observed Packet Stream from which packets are selected disproportionately (e.g. to evade selection, or overload measurement systems).

(f) Parallel Metering Processes: the protocol must support simultaneous operation of multiple independent Metering Processes at the same host.

(g) Causality: the selection decision for each packet should depend only weakly, if at all, upon future packets arrivals. This promotes ubiquity by limiting the complexity of the selection logic.

(h) Encrypted Packets: Selectors that interpret packet fields must be configurable to ignore (i.e. not select) encrypted packets, when they are detected.

Specific Selectors are outlined in Section 5, and described in more detail in the companion document [PSAMP-TECH].

4.2 Generic Reporting Requirements

(i) Self-defining: the Report Stream must be complete in the sense that no additional information need be retrieved from the Observation Point in order to interpret and analyze the reports.

(j) Indication of Information Loss: the Report Stream must include sufficient information to indicate or allow the detection of loss occurring within the Selection, Metering, and/or Exporting Processes, or in transport. This may be achieved by the use of sequence numbers.

(k) Accuracy: the Report Stream must include information that enables the accuracy of measurements to be determined.

(l) Faithfulness: all reported quantities that relate to the packet treatment must reflect the router state and configuration encountered by the packet at the time it is received by the Metering Process.

(m) Privacy: although selection of the content of Packet Reports must be responsive to the needs of measurement applications, it must also conform with [RFC-2804]. In

particular, full packet capture of arbitrary packet streams is explicitly out of scope.

See section 6 for further discussions on Reporting.

4.3 Generic Exporting Process Requirements

(n) Timeliness: configuration must allow for limiting of buffering delays for the formation and transmission for Export Packets. See Section 8.5 for further details.

(o) Congestion Avoidance: export of a Report Stream across a network must be congestion avoiding in compliance with [RFC-2914]. This is discussed further in Section 8.3.

(p) Secure Export:

(i) confidentiality: the option to encrypt exported data must be provided.

(ii) integrity: alterations in transit to exported data must be detectable at the Collector

(iii) authenticity: authenticity of exported data must be verifiable by the Collector in order to detect forged data.

The motivation here is the same as for security in IPFIX export; see Sections 6.3 and 10 of [RFC-3917].

4.4 Generic Configuration Requirements

(q) Ease of Configuration: of sampling and export parameters, e.g. for automated remote reconfiguration in response to collected reports.

(r) Secure Configuration: the option to configure via protocols that prevent unauthorized reconfiguration or eavesdropping on configuration communications must be available. Eavesdropping on configuration might allow an attacker to gain knowledge that would be helpful in crafting a packet stream to evade subversion, or overload the measurement infrastructure.

Configuration is discussed in Section 9.

5. Packet Selection

This section details specific requirements for the Selection Process, motivated by the generic requirements of Section 3.3.

5.1 Two Types of Selector

PSAMP categorizes selectors into two types:

* Filtering: a filter is a Selector that selects a packet deterministically based on the Packet Content, or its treatment, or functions of these occurring in the Selection State. Two examples are:

  (i) Property match filtering: a packet is selected if a specific field in the packet equals a predefined value.

  (ii) Hash-based selection: a hash function is applied to the Packet Content, and the packet is selected if the result falls in a specified range.

* Sampling: a selector that is not a filter is called a sampling operation. This reflects the intuitive notion that if the selection of a packet cannot be determined from its content alone, there must be some type of sampling taking place.

  Sampling operations can be divided into two subtypes:

  (i) Content-independent sampling, which does not use Packet Content in reaching sampling decisions. Examples include systematic sampling, and uniform pseudorandom sampling driven by a pseudorandom number whose generation is independent of Packet Content. Note that in Content-independent Sampling it is not necessary to access the Packet Content in order to make the selection decision.

  (ii) Content-dependent sampling, in which the Packet Content is used in reaching selection decisions. An application is pseudorandom selection according to a probability that depends on the contents of a packet field, e.g., sampling packets with a probability dependent on their TCP/UDP port numbers. Note that this is not a Filter.

5.2 PSAMP Packet Selectors

A spectrum of packet selectors is described in detail in [PSAMP-TECH]. Here we only briefly summarize the meanings for completeness.

A PSAMP Selection Process must support at least one of the following Selectors.

* systematic count based sampling: packet selection is triggered periodically by packet count, a number of successive packets being selected subsequent to each trigger.

* systematic time based sampling: similar to systematic count based except that selection is reckoned with respect to time rather than count. Packet selection is triggered at periodic

instants separated by a time called the spacing. All packets that arrive within a certain time of the trigger (called the interval length) are selected.

* probabilistic n-out-of-N sampling: from each count-based successive block of N packets, n are selected at random.

* uniform probabilistic sampling: packets are selected independently with fixed sampling probability p.

* non-uniform probabilistic sampling: packets are selected independently with probability p that depends on Packet Content.

* property match filtering

With this Filtering method a packet is selected if a specific field within the packet and/or on properties of the router state equal(s) a predefined value. Possible filter fields are all IPFIX flow attributes specified in [RFC-5102]. Further fields can be defined by vendor specific extensions.

A packet is selected if Field=Value. Masks and ranges are only supported to the extent to which [RFC-5102] allows them e.g. by providing explicit fields like the netmasks for source and destination addresses.

AND operations are possible by concatenating filters, thus producing a composite selection operation. In this case, the ordering in which the filtering happens is implicitly defined (outer filters come after inner filters). However, as long as the concatenation is on filters only, the result of the cascaded filter is independent from the order, but the order may be important for implementation purposes, as the first filter will have to work at a higher rate. In any case, an implementation is not constrained to respect the filter ordering, as long as the result is the same, and it may even implement the composite filtering in filtering in one single step.

OR operations are not supported with this basic model. More sophisticated filters (e.g. supporting bitmasks, ranges or OR operations etc.) can be realized as vendor specific schemes.

Property match operations should be available for different protocol portions of the packet header:

(i) the IP header (excluding options in IPv4, stacked headers in IPv6)

(ii) transport header

(iii) encapsulation headers (e.g. the MPLS label stack, if present)

When the PSAMP Device offers property match filtering, and, in its usual capacity other than in performing PSAMP functions, identifies or processes information from IP, transport or encapsulation protocols, then the information should be made available for filtering. For example, when a PSAMP Device is a router that routes based on destination IP address, that field should be made available for filtering. Conversely, a PSAMP Device that does not route is not expected to be able to locate an IP address within a packet, or make it available for Filtering, although it may do so.

Since packet encryption alters the meaning of encrypted fields, property match filtering must be configurable to ignore encrypted packets, when detected.

The Selection Process may support filtering based on the properties of the router state:

(i) Ingress interface at which packet arrives equals a specified value

(ii) Egress interface to which packet is routed to equals a specified value

(iii) Packet violated Access Control List (ACL) on the router

(iv) Failed Reverse Path Forwarding (RPF). Packets that match the Failed Reverse Path Forwarding (RPF) condition are packets for which ingress filtering failed as defined in [RFC3704].

(v) Failed Resource Reservation (RSVP). Packets that match the Failed Resource Reservation condition are packets that do not fulfill the RSVP specification as defined in [RFC-2205].

(vi) No route found for the packet

(vii) Origin Border Gateway Protocol (BGP) Autonomous System (AS) [RFC-4271] equals a specified value or lies within a given range

(viii) Destination BGP AS equals a specified value or lies within a given range

Router architectural considerations may preclude some information concerning the packet treatment being available at line rate for selection of packets. For example, the Selection

Process may not be implemented in the fast path that is able to access routing state at line rate. However, when filtering follows sampling (or some other selection operation) in a Composite Selector, the rate of the Packet Stream output from the sampler and input to the filter may be sufficiently slow that the filter could select based on routing state.

* Hash-based Selection:

  Hash-based selection will employ one or more hash functions to be standardized. A hash function is applied to a subset of Packet Content, and the packet is selected of the resulting hash falls in a specified range. The stronger the hash function, the more closely hash-based selection approximates uniform random sampling. Privacy of hash selection range and hash function parameters obstructs subversion of the selector by packets that are crafted either to avoid selection or to be selected. Privacy of the hash function is not required. Robustness and security considerations of hash-based selection are further discussed in further in [PSAMP-TECH]. Applications of hash-based sampling are described in Section 11.

5.3 Selection Fraction Terminology

* Population:

  A population is a Packet Stream, or a subset of a Packet Stream. A Population can be considered as a base set from which packets are selected. An example is all packets in the Observed Packet Stream that are observed within some specified time interval.

* Population Size:

  The Population Size is the number of all packets in a Population.

* Configured Selection Fraction

  The Configured Selection Fraction is the ratio of the number of packets selected by a Selector from an input Population, to the Population Size, as based on the configured selection parameters.

* Attained Selection Fraction

  The Attained Selection Fraction is the actual ratio of the number of packets selected by a Selector from an input

Population, to the Population Size.

For some sampling methods the Attained Selection Fraction can differ from the Configured Selection Fraction due to, for example, the inherent statistical variability in sampling decisions of probabilistic sampling and hash-based selection. Nevertheless, for large Population Sizes and properly configured Selectors, the Attained Selection Fraction usually approaches the Configured Selection Fraction.

The notions of Configured/Attained Selection Fraction extend beyond Selectors. An illustrative example is the Configured Selection Fraction of the composition of the Metering Process with the Exporting Process. Here the Population is the Observed Packet Stream or a subset thereof. The Configured Selection Fraction is the fraction of the Population for which Packet Reports which are expected to reach the Collector. This quantity may reflect additional parameters, not necessarily described in the PSAMP protocol, that determine the degree of loss suffered by Packet Reports en route to the Collector, e.g., the transmission bandwidth available to the Exporting Process. In this example, the Attained Selection Fraction is the fraction of Population packets for which reports did actually reach the Collector, and thus incorporates the effect of any loss of Packet Reports due, e.g, to resource contention at the Observation Point, or during transmission.

5.4 Input Sequence Numbers for Primitive Selectors

Each instance of a Primitive Selector must maintain a count of packets presented at its input. The counter value is to be included as a sequence number for selected packets. The sequence numbers are considered as part of the packet's Selection State.

Use of input sequence numbers enables applications to determine the Attained Selection Fraction, and hence correctly normalize network usage estimates regardless of loss of information, regardless of whether this loss occurs because of discard of packet reports in the Metering Process (e.g. due to resource contention in the host of these processes), or loss of export packets in transmission or collection. See [RFC-3176] for further details.

As an example, consider a set of n consecutive packet reports r1, r2,... , rn, selected by a sampling operation and received at a Collector. Let s1, s2,..., sn be the input sequence numbers reported by the packets. The Attained Selection Fraction for the composite of the measurement and exporting processes, taking into account both packet sampling at the Observation Point and loss in

transmission, is computed as R = (n-1)/(sn-s1). (Note R would be 1 if all packets were selected and there were no transmission loss).

The Attained Selection Fraction can be used to estimate the number of bytes present in a portion of the Observed Packet Stream. Let b1, b2,..., bn be the number of bytes reported in each of the packets that reached the Collector, and set B = b1+b2+...+bn. Then the total bytes present in packets in the Observed Packet Stream whose input sequence numbers lie between s1 and sn is estimated by B/R, i.e, scaling up the measured bytes through division by the Attained Selection Fraction

With Composite Selectors, an input sequence number must be reported for each Selector in the composition.

5.5 Composite Selectors

The ability to compose Selectors in a Selection Process should be provided. The following combinations appear to be most useful for applications:

* concatentation of property match filters. This is useful for constructing the AND of the component filters.

* filtering followed by sampling.

* sampling followed by filtering.

Composite Selectors are useful for drill down applications. The first component of a composite selector can be used to reduce the load on the second component. In this setting, the advantage to be gained from a given ordering can depends on the composition of the packet stream.

5.6 Constraints on the Selection Fraction

Sampling at full line rate, i.e. with probability 1, is not excluded in principle, although resource constraints may not permit it in practice.

6. Reporting

This section details specific requirements for reporting, motivated by the generic requirements of Section 3.4

6.1 Mandatory Contents of Packet Reports: Basic Reports

Packet Reports must include the following:

(i) the input sequence number(s) of any Selectors that acted on the packet in the instance of a Metering Process which produced the report.

(ii) the identifier of the Metering Process that produced the selected packet

The Metering Process must support inclusion of the following in each Packet Report, as a configurable option:

(iii) a basic report on the packet, i.e., some number of contiguous bytes from the start of the packet, including the packet header (which includes network layer and any encapsulation headers) and some subsequent bytes of the packet payload.

Some devices may not have the resource capacity or functionality to provide more detailed packet reports than those in (i), (ii) and (iii) above. Using this minimum required reporting functionality, the Metering Process places the burden of interpretation on the Collector, or on applications that it supplies. Some devices may have the capability to provide extended packet reports, described in the next section.

6.2 Extended Packet Reports

The Metering Process may support inclusion in Packet Reports of the following information, inclusion any or all being configurable as an option.

(iv) fields relating to the following protocols used in the packet: IPv4, IPV6, transport protocols, and encapsulation protocols including MPLS

(v) packet treatment, including:

- identifiers for any input and output interfaces of the Observation Point that were traversed by the packet

- source and destination BGP AS

(vi) Selection State associated with the packet, including:

- the timestamp of observation of the packet at the Observation Point. The timestamp should be reported to microsecond resolution.

- hashes, where calculated.

It is envisaged that selection of fields for Extended Packet Reporting may be used to reduce reporting bandwidth, in which case the option to report information in (iii) may not be exercised.

6.3 Extended Packet Reports in the Presence of IPFIX

If an IPFIX metering process is supported at the Observation Point, then in order to be PSAMP compliant, Extended Packet

Reports must be able to include all fields required in the IPFIX information model [RFC-5102], with modifications appropriate to reporting on single packets rather than flows.

6.4 Report Interpretation

The Report Interpretation must include:

(i) configuration parameters of the Selectors of the packets reported on.

(ii) format of the Packet Report;

(iii) indication of the inherent accuracy of the reported quantities, e.g., of the packet timestamp.

The accuracy measure in (iii) is of fundamental importance for estimating the likely error attached to estimates formed from the Packet Reports by applications.

The requirements for robustness and transparency are motivations for including Report Interpretation in the Report Stream: it makes the Report Stream self-defining. The PSAMP framework excludes reliance on an alternative model in which interpretation is recovered out of band. This latter approach is not robust with respect to undocumented changes in Selector configuration, and may give rise to future architectural problems for network management systems to coherently manage both configuration and data collection.

It is not envisaged that all Report Interpretation be included in every Packet Report. Many of the quantities listed above are expected to be relatively static; they could be communicated periodically, and upon change.

7. Parallel Metering Processes

Because of the increasing number of distinct measurement applications, with varying requirements, it is desirable to set up parallel Metering Processes on a given Observed Packet Stream. A device capable of hosting a Metering Process should be able to support more than one independently configurable Metering Process simultaneously. Each such Metering Process should have the option of being equipped with its own Exporting Process; otherwise the parallel Metering Processes may share the same Exporting Process.

Each of the parallel Metering Processes should be independent. However, resource constraints may prevent complete reporting on a

packet selected by multiple Selection Processes. In this case, reporting for the packet must be complete for at least one Metering Process; other Metering Processes need only record that

they selected the packet, e.g., by incrementing a counter. The priority amongst Metering Processes under resource contention should be configurable.

It is not proposed to standardize the number of parallel Metering Processes.

8. Exporting Process

This section details specific requirements for the Exporting Process, motivated by the generic requirements of Section 3.6

8.1 Use of IPFIX

PSAMP will use the IP Flow Information eXport (IPFIX) protocol for export of the Report Stream. The IPFIX protocol is well suited for this purpose, because the IPFIX architecture matches the PSAMP architecture very well and the means provided by the IPFIX protocol are sufficient for PSAMP purposes. On the other hand, not all features of the IPFIX protocol will need to be implemented by some PSAMP devices. For example, a device that offers only content-independent sampling and basic PSAMP reporting has no need to support IPFIX capabilities based on packet fields.

8.2 Export Packets

Export packets may contain one or more Packet Reports, and/or Report Interpretation. Export packets must also contain:

(i) An identifier for the Exporting Process

(ii) An export packet sequence number.

An export packet sequence number enables the Collector to identify loss of export packets in transit. Note that some transport protocols, e.g. UDP, do not provide sequence numbers. Moreover, having sequence numbers available at the application level enables the Collector to calculate packet loss rate for use, e.g., in estimating original traffic volumes from export packet that reach the Collector.

8.3 Congestion-aware Unreliable Transport

The export of the Report Stream does not require reliable export. Section 5.4 shows that the use of input sequence numbers in packet Selectors means that the ability to estimate traffic rates is not impaired by export loss. Export packet loss becomes another form of sampling, albeit a less desirable, and less

controlled, form of sampling.

In distinction, retransmission of lost Export Packets consumes additional network resources. The requirement to store unacknowledged data is an impediment to having ubiquitous support for PSAMP.

In order to jointly satisfy the timeliness and congestion avoidance requirements of Section 4.3, a congestion-aware unreliable transport protocol may be used. IPFIX is compatible with this requirement, since it mandates support of the Stream Control Transmission Protocol (SCTP) [RFC-4960] and the SCTP Partial Reliability Extension [RFC-3758].

IPFIX also allows the use of User Datagram Protocol (UDP) [RFC-768] although it is not a congestion-aware protocol. However, in this case, the Export Packets must remain wholly within the administrative domains of the operators [RFC-5101]. The PSAMP exporting process is equipped with a configurable export rate limit (see Section 8.4 following) that can be used to limit the export rate when a congestion aware transport protocol is not used. The Collector, upon detection of export packet loss through missing export sequence numbers, may reconfigure the export rate limit downwards in order to avoid congestion.

8.4 Configurable Export Rate Limit

The exporting process must have an export rate limit, configurable per Exporting Process. This is useful for two reasons:

(i) Even without network congestion, the rate of packet selection may exceed the capacity of the Collector to process reports, particularly when many Exporting Processes feed a common Collector. Use of an Export Rate Limit allows control of the global input rate to the Collector.

(ii) IPFIX provides export using UDP as the transport protocol in some circumstances. An Export Rate Limit allows the capping of the export rate to match both path link speeds and the capacity of the Collector.

8.5 Limiting Delay for Export Packets

Low measurement latency allows the traffic monitoring system to be more responsive to real-time network events, for example, in quickly identifying sources of congestion. Timeliness is generally a good thing for devices performing the sampling since it minimizes the amount of memory needed to buffer samples.

Keeping the packet dispatching delay small has other benefits

besides limiting buffer requirements. For many applications a resolution of 1 second is sufficient. Applications in this category would include: identifying sources associated with

congestion, tracing denial of service attacks through the network, and constructing traffic matrices. Furthermore, keeping dispatch delay within the resolution required by applications eliminates the need for timestamping by synchronized clocks at observation points, or for the Observation Points and Collector to maintain bi-directional communication in order to track clock offsets. The Collector can simply process Packet Reports in the order that they are received, using its own clock as a "global" time base. This avoids the complexity of buffering and reordering samples. See [DuGeGr02] for an example.

The delay between observation of a packet and transmission of a Export Packet containing a report on that packet has several components. It is difficult to standardize a given numerical delay requirement, since in practice the delay may be sensitive to processor load at the Observation Point. Therefore, PSAMP aims to control that portion of the delay within the Observation Point that is due to buffering in the formation and transmission of Export Packets.

In order to limit delay in the formation of Export Packets, the Exporting Process must provide the ability to close out and enqueue for transmission any Export Packet during formation as soon as it includes one Packet Report.

In order to limit the delay in the transmission of Export Packets, a configurable upper bound to the delay of an Export Packet prior to transmission must be provided. If the bound is exceeded the Export Packet is dropped. This functionality can be provided by the timed reliability service of the SCTP Partial Reliability Extension [RFC-3758].

The Exporting Process may enqueue the Report Stream in order to export multiple Packet Reports in a single export packet. Any consequent delay must still allow for timely availability of Packet Reports as just described. The timed reliability service of the SCTP Partial Reliability Extension [RFC-3758] allows the dropping of packets from the export buffer once their age in the buffer exceeds a configurable bound. A suitable default value for the bound should be used in order to avoid a low transmission rate due to misconfiguration.

8.6 Export Packet Compression

To conserve network bandwidth and resources at the Collector, the Export Packets may be compressed before export. Compression is expected to be quite effective since the sampled packets may share many fields in common, e.g. if a filter focuses on packets

with certain values in particular header fields. Using compression, however, could impact the timeliness of Packet Reports. Any consequent delay must not violate the timeliness requirement for availability of Packet Reports at the Collector.

8.7 Collector Destination

When exporting to a remote Collector, the Collector is identified by IP address, transport protocol, and transport port number.

8.8 Local Export

The Report Stream may be directly exported to on-board measurement based applications, for example those that form composite statistics from more than one packet. Local export may be presented through an interface direct to the higher level applications, i.e., through an API, rather than employing the transport used for off-board export. Specification of such an API is outside the scope of the PSAMP framework.

A possible example of Local Export could be that packets selected by the PSAMP Metering Process serve as the input for the IPFIX protocol, which then forms flow records out of the stream of selected packets.

9. Configuration and Management

A key requirement for PSAMP is the easy reconfiguration of the parameters of the Metering Process, including those for selection and packet reports, and of the Exporting Process. An important example is to support measurement-based applications that want to adaptively drill-down on traffic detail in real-time.

To facilitate retrieval and monitoring of parameters, they are to reside in a Management Information Base (MIB). Mandatory monitoring objects will cover all mandatory PSAMP functionality. Alarming of specific parameters could be triggered with thresholding mechanisms such as the RMON event and alarm [RFC-2819] or the event MIB [RFC-2981].

For configuring parameters of the Metering Process, several alternatives are available including a MIB module with writeable objects, as well as other configuration protocols. For configuring parameters of the Exporting Process, the Packet Report, and the Report Interpretation, which is an IFPIX task, the IPFIX configuration method(s) should be used.

Although management and configuration of collectors is out of scope, a PSAMP device, to the extent that it employs IPFIX as an export protocol, inherits from IPFIX the capability to detect and

recover from collector failure; see Section 8.2 of [IPFIX-ARCH].

10. Feasibility and Complexity

In order for PSAMP to be supported across the entire spectrum of networking equipment, it must be simple and inexpensive to implement. One can envision easy-to-implement instances of the mechanisms described within this draft. Thus, for that subset of instances, it should be straightforward for virtually all system vendors to include them within their products. Indeed, sampling and filtering operations are already realized in available equipment.

Here we give some specific arguments to demonstrate feasibility and comment on the complexity of hardware implementations. We stress here that the point of these arguments is not to favor or recommend any particular implementation, or to suggest a path for standardization, but rather to demonstrate that the set of possible implementations is not empty.

10.1 Feasibility

10.1.1 Filtering

Filtering consists of a small number of mask (bit-wise logical), comparison and range (greater than) operations. Implementation of at least a small number of such operations is straightforward. For example, filters for security access control lists (ACLs) are widely implemented. This could be as simple as an exact match on certain fields, or involve more complex comparisons and ranges.

10.1.2 Sampling

Sampling based on either counters (counter set, decrement, test for equal to zero) or range matching on the hash of a packet (greater than) is possible given a small number of selectors, although there may be some differences in ease of implementation for hardware vs. software platforms.

10.1.3 Hashing

Hashing functions vary greatly in complexity. Execution of a small number of sufficient simple hash functions is implementable at line rate. Concerning the input to the hash function, hop-invariant IP header fields (IP address, IP identification) and TCP/UDP header fields (port numbers, TCP sequence number) drawn from the first 40 bytes of the packet have been found to possess a considerable variability; see [DuGr01].

10.1.4 Reporting

The simplest Packet Report would duplicate the first n bytes of the packet. However, such an uncompressed format may tax the bandwidth available to the Exporting Process for high sampling rates; reporting selected fields would save on this bandwidth. Thus there is a trade-off between simplicity and bandwidth limitations.

10.1.5 Exporting

Ease of exporting export packets depends on the system architecture. Most systems should be able to support export by insertion of export packets, even through the software path.

10.2 Potential Hardware Complexity

Achieving low constants for performance while minimizing hardware resources is, of course, a challenge, especially at very high clock frequencies. Most of the Selectors, however, are very basic and their implementations very well understood; in fact, the average Application Specific Integrated Circuit (ASIC) designer simply uses canned library instances of these operations rather than design them from scratch. In addition, networking equipment generally does not need to run at the fastest clock rates, further reducing the effort required to get reasonably efficient implementations.

Simple bit-wise logical operations are easy to implement in hardware. Such operations (NAND/NOR/XNOR/NOT) directly translate to four-transistor gates. Each bit of a multiple-bit logical operation is completely independent and thus can be performed in parallel incurring no additional performance cost above a single bit operation.

Comparisons (EQ/NEQ) take O(log(M)) stages of logic, where M is the number of bits involved in the comparison. The log(M) is required to accumulate the result into a single bit.

Greater than operations, as used to determine whether a hash falls in a selection range, are a determination of the most significant not-equivalent bit in the two operands. The operand with that most-significant-not-equal bit set to be one is greater than the other. Thus, a greater than operation is also an O(log(M)) stages of logic operation. Optimized implementations of arithmetic operations are also O(log(M)) due to propagation of the carry bit.

Setting a counter is simply loading a register with a state. Such an operation is simple and fast O(1). Incrementing or decrementing a counter is a read, followed by an arithmetic

operation followed by a store. Making the register dual-ported does take additional space, but it is a well-understood

technique. Thus, the increment/decrement is also an O(log(M)) operation.

Hashing functions come in a variety of forms. The computation involved in a standard Cyclic Redundancy Code (CRC) for example are essentially a set of XOR operations, where the intermediate result is stored and XORed with the next chunk of data. There are only O(1) operations and no log complexity operations. Thus, a simple hash function, such as CRC or generalizations thereof, can be implemented in hardware very efficiently.

At the other end of the range of complexity, the MD5 function uses a large number of bit-wise conditional operations and arithmetic operations. The former are O(1) operations and the latter are O(log(M)). MD5 specifies 256 32b ADD operations per 16B of input processed. Consider processing 10Gb/sec at 100MHz (this processing rate appears to be currently available). This requires processing 12.5B/cycle, and hence at least 200 adders, a sizeable number. Because of data dependencies within the MD5 algorithm, the adders cannot be simply run in parallel, thus requiring either faster clock rates and/or more advanced architectures. Thus, selection hashing functions as complex as MD5 may be precluded for ubiquitous use at full line rate. This motivates exploring the use of selection hash functions with complexity somewhere between that of MD5 and CRC. In some applications (see Section below) a second hash may be calculated on only selected packets; MD5 is feasible for this purpose if the rate of production of selected packets is sufficiently low.

11. Applications

We first describe several representative operational applications that require traffic measurements at various levels of temporal and spatial granularity. Some of the goals here appear similar to those of IPFIX, at least in the broad classes of applications supported. The major benefit of PSAMP is the support of new network management applications, specifically, those enabled by the packet Selectors that it supports.

11.1 Baseline Measurement and Drill Down

Packet sampling is ideally suited to determine the composition of the traffic across a network. The approach is to enable measurement on a cut-set of the network links such that each packet entering the network is seen at least once, for example, on all ingress links. Unfiltered sampling with a relatively low selection fraction establishes baseline measurements of the network traffic. Packet Reports include packet attributes of

common interest: source and destination address and port numbers, prefix, protocol number, type of service, etc. Traffic matrices are indicated by reporting source and destination AS matrices. Absolute traffic volumes are estimated by renormalizing the

sampled traffic volumes through division by either the Configured Selection Fraction, or by the Attained Selection Fraction (as derived from input packet counters included in the Report Stream)

Suppose an operator or a measurement-based application detects an interesting subset of a Packet Stream, as identified by a particular packet attribute. Real-time drill-down to that subset is achieved by instantiating a new Metering Process on the same Observed Packet Stream from which the subset was reported. The Selection Process of the new Metering Process filters according to the attribute of interest, and composes with sampling if necessary to manage the attained fraction of packets selected.

11.2 Trajectory Sampling

The goal of trajectory sampling is the selection of a subset of packets at all enabled Observation Points at which they are observed in a network domain. Thus the selection decisions are consistent in the sense that each packet is selected either at all enabled Observation Points, or at none of them. Trajectory sampling is realized by hash-based selection if all enabled Observation Points apply a common hash function to a portion of the Packet Content that is invariant along the packet path. (Thus, fields such at TTL and CRC are excluded).

The trajectory followed by a packet is reconstructed from Packet Reports on it that reach the Collector. Reports on a given packet are associated either by matching a label comprising the invariant reported Packet Content, or possibly some digest of it. The reconstruction of trajectories, and methods for dealing with possible ambiguities due to label collisions (identical labels reported by different packets) and potential loss of reports in transmission are dealt with in [DuGr01], [DuGeGr02] and [DuGr04].

11.3 Passive Performance Measurement

Trajectory sampling enables the tracking of the performance experience by customer traffic, customers identified by a list of source or destination prefixes, or by ingress or egress interfaces. Operational uses include the verification of Service Level Agreements (SLAs), and troubleshooting following a customer complaint.

In this application, trajectory sampling is enabled at all network ingress and egress interfaces. Rates of loss in transit between ingress and egress are estimated from the proportion of trajectories for which no egress report is received. Note that

loss of customer packets is distinguishable from loss of packet reports through use of report sequence numbers. Assuming

synchronization of clocks between different entities, delay of customer traffic across the network may also be measured; see [Zs02].

Extending hash-selection to all interfaces in the network would enable attribution of poor performance to individual network links.

11.4 Troubleshooting

PSAMP Packet Reports can also be used to diagnose problems whose occurrence is evident from aggregate statistics, per interface utilization and packet loss statistics. These statistics are typically moving averages over relatively long time windows, e.g., 5 minutes, and serve as a coarse-grain indication of operational health of the network. The most common method of obtaining such measurements are through the appropriate SNMP MIBs (MIB-II [RFC-1213] and vendor-specific MIBs.)

Suppose an operator detects a link that is persistently overloaded and experiences significant packet drop rates. There is a wide range of potential causes: routing parameters (e.g., OSPF link weights) that are poorly adapted to the traffic matrix, e.g., because of a shift in that matrix; a denial of service attack or a flash crowd; a routing problem (link flapping). In most cases, aggregate link statistics are not sufficient to distinguish between such causes, and to decide on an appropriate corrective action. For example, if routing over two links is unstable, and the links flap between being overloaded and inactive, this might be averaged out in a 5 minute window, indicating moderate loads on both links.

Baseline PSAMP measurement of the congested link, as described in Section 11.1, enables measurements that are fine grained in both space and time. The operator has to be able to determine how many bytes/packets are generated for each source/destination address, port number, and prefix, or other attributes, such as protocol number, MPLS forwarding equivalence class (FEC), type of service, etc. This allows the precise determination of the nature of the offending traffic. For example, in the case of a Distributed Denial of Service(DDoS) attack, the operator would see a significant fraction of traffic with an identical destination address.

In certain circumstances, precise information about the spatial flow of traffic through the network domain is required to detect and diagnose problems and verify correct network behavior. In the case of the overloaded link, it would be very helpful to know

the precise set of paths that packets traversing this link follow. This would readily reveal a routing problem such as a loop, or a link with a misconfigured weight. More generally, complex diagnosis scenarios can benefit from measurement of

traffic intensities (and other attributes) over a set of paths that is constrained in some way. For example, if a multihomed customer complains about performance problems on one of the access links from a particular source address prefix, the operator should be able to examine in detail the traffic from that source prefix which also traverses the specified access link towards the customer.

While it is in principle possible to obtain the spatial flow of traffic through auxiliary network state information, e.g., by downloading routing and forwarding tables from routers, this information is often unreliable, outdated, voluminous, and contingent on a network model. For operational purposes, a direct observation of traffic flow provided by trajectory sampling is more reliable, as it does not depend on any such auxiliary information. For example, if there was a bug in a router's software, direct observation would allow the diagnosis the effect of this bug, while an indirect method would not.

12. Security Considerations

12.1 Relation of PSAMP and IPFIX Security for Exporting Process

As detailed in Section 4.3, PSAMP shares with IPFIX security requirements for export, namely, confidentiality, integrity and authenticity of the exported data; see also Sections 6.3 and 10 of [RFC-3917]. Since PSAMP will use IPFIX for export, it can employ the IPFIX protocol [RFC-5101] to meet its requirements.

12.2 PSAMP Specific Privacy Considerations

In distinction with IPFIX, a PSAMP device may, in some configurations, report some number of initial bytes of the packet, which may include some part of a packet payload. This option is conformant with the requirements of [RFC-2804] since it does not mandate configurations that would enable capture of an entire packet stream of a flow: neither a unit sampling rate (1 in 1 sampling) nor reporting a specific number of initial bytes, are required by the PSAMP protocol.

To preserve privacy of any users acting as sender or receiver of the observed traffic the contents of the packet reports must be able to remain confidential in transit between the exporting PSAMP device and the collector. PSAMP will use IPFIX as the exporting protocol, and the IPFIX protocol must provide mechanisms to ensure confidentiality of the exporting process, for example, encryption of export packets [RFC-5101].

12.3 Security Considerations for Hash-Based Selection

12.3.1 Modes and Impact of vulnerabilities

A concern for Hash-based Selection is whether some large set of related packets could be disproportionately sampled, either

(i) through unanticipated behavior in the Hash Function, or
(ii) because the packets had been deliberately crafted to have this property.

As detailed below, only cryptographic hash functions (e.g. one based on MD5) employing a private parameter are sufficiently strong to withstand the range of conceivable attacks. However, implementation considerations may preclude operating the strongest hash functions at line rate. For this reason PSAMP is not expected to standardize around a cryptographic hash function at the present time. The purpose of this section is to inform discussion of the vulnerabilities and trade-offs associated with different hash function choices. Section 6.2.2 of [PSAMP-TECH] does this in more detail.

An attacker able to predict packet sampling outcomes could craft a packet stream that could evade selection; or another that could overwhelm the measurement infrastructure with all its packets being selected. An attacker may attempt to do this based on knowledge of the hash function. An attacker could employ knowledge of selection outcomes of a known packet stream to reverse engineer parameters of the hash function. This knowledge could be gathered e.g. from billing information, reactions of intrusion detection systems, or observation of a report stream.

Since hash-based selection is deterministic, it is vulnerable to replay attacks. Repetition of a single packet may be noticeable to other measurement methods if employed (e.g. collection of flow statistics), whereas a set of distinct packets that appears statistically similar to regular traffic may be less noticeable. The impact of replay attacks on hash based selection may be mitigated by repeated changing of hash function parameters.
.

12.3.2 Use of Private Parameters in Hash Functions

Because hash functions for Hash-based selection are to be standardized and hence public, the packet selection decision must be controlled by some private quantity associated with the hash-based Selector. Making private the range of hash values for which packets are selected is not alone sufficient to prevent an attacker crafting a stream of distinct packets that are

disproportionately selected. A private parameter must be used within the hash function, for example, a private modulus in a

hash function, or by concatenating the hash input with a private string prior to hashing.

12.3.3 Strength of Hash Functions

The specific choice of hash function and it usage determines the types of potential vulnerability:

* Cryptographic hash functions: when a private parameter is used, future selection outcomes cannot be predicted even by an attacker with knowledge of past selection outcomes.

* Non-cryptographic hash functions:

  Using knowledge of past selection outcomes: some well known hash functions, e.g., CRC-32, are vulnerable to attacks, in the sense that their private parameter can be determined with knowledge of sufficiently many past selections, even when a private parameter is used; see [GoRe07].

  No knowledge of past selection outcomes: using a private parameter hardened the hash function to classes of attacks that work when the parameter is public, although vulnerability to future attacks is not precluded.

12.4 Security Guidelines for Configuring PSAMP

Hash-function parameters configured in a PSAMP device are sensitive information, which must be kept private. As well as using probing techniques to discover parameters of non-cryptographic hash functions as described above, implementation and procedural weaknesses may lead to attackers discovering parameters, whatever class of hash function is used. The following measures may prevent this from occurring:

Hash function parameters must not be displayable in cleartext on PSAMP devices. This reduces the chance for the parameters to be discovered by unauthorized access to the PSAMP device.

Hash function parameters must not be remotely set in cleartext over a channel which may be eavesdropped.

Hash function parameters must be changed regularly. Note that such changes must be synchronized over all PSAMP devices in a domain under which Trajectory Sampling is employed in order to maintain consistent sampling of packets over the domain.

Default hash function parameter values should be initialized randomly, in order to avoid predictable values that attackers

could exploit.

13. IANA Considerations

This document has no actions for IANA.

14. References

14.1 Normative References

[PSAMP-PROTO] B. Claise (Ed.) Packet Sampling (PSAMP) Protocol Specifications, RFC XXXX. [Currently Internet Draft draft-ietf-psamp-protocol-09.txt, work in progress, December 2007.]

[PSAMP-INFO] T. Dietz, F. Dressler, G. Carle, B. Claise, Information Model for Packet Sampling Exports, RFC XXXX. [Currently Internet Draft, draft-ietf-psamp-info-08, February 2008.]

[RFC-5101] B. Claise (Ed.) "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information'', RFC 5101, January 2008.

[RFC-0791] J. Postel, "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC-5102] J. Quittek, S. Bryant, B. Claise, P. Aitken, J. Meyer, "Information Model for IP Flow Information Export", RFC 5102, January 2008.

[RFC-4960] R. Stewart, (ed.) "Stream Control Transmission Protocol", RFC 4960, September 2007.

[RFC-3758] R. Stewart, M. Ramalho, Q. Xie, M. Tuexen, P. Conrad, "SCTP Partial Reliability Extension", RFC 3758, May 2004.

[PSAMP-TECH] T. Zseby, M. Molina, F. Raspall, N. G. Duffield, S. Niccolini, Sampling and Filtering Techniques for IP Packet Selection, RFC XXXX. [Currently Internet Draft, draft-ietf-psamp-sample-tech-10.txt, work in progress, July 2005.

14.2 Informative References

[RFC3704] F. Baker, P. Savola, Ingress Filtering for
Multihomed Networks, RFC3704, March 2004.

[RFC-2205] R. Braden (Ed.), L. Zhang, S. Berson, S. Herzog, S. Jamin, Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification, RFC2205, September 1997.

[RFC-2460] S. Deering, R. Hinden, Internet Protocol, Version 6 (IPv6) Specification, RFC 2460, December 1998.

[DuGr01] N. G. Duffield and M. Grossglauser, Trajectory Sampling for Direct Traffic Observation, IEEE/ACM Trans. on Networking, 9(3), 280-292, June 2001.

[DuGeGr02] N.G. Duffield, A. Gerber, M. Grossglauser, Trajectory Engine: A Backend for Trajectory Sampling, IEEE Network Operations and Management Symposium 2002, Florence, Italy, April 15-19, 2002.

[DuGr04] N. G. Duffield and M. Grossglauser, Trajectory Sampling with Unreliable Reporting, Proc IEEE Infocom 2004, Hong Kong, March 2004,

[RFC-2914] S. Floyd, Congestion Control Principles, RFC 2914, September 2000.

[GoRe07] S. Goldberg, J. Rexford, "Security Vulnerabilities and Solutions for Packet Sampling", IEEE Sarnoff Symposium, Princeton, NJ, May 2007.

[RFC-2804] IAB and IESG, Network Working Group, IETF Policy on Wiretapping, RFC 2804, May 2000

[RFC-2981] R. Kavasseri, Ed., ''Event MIB'', RFC 2981, October 2000.

[RFC-1213] K. McCloghrie, M. Rose, Management Information Base for Network Management of TCP/IP-based internets:MIB-II, RFC 1213, March 1991.

[RFC-3176] P. Phaal, S. Panchen, N. McKee, InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks, RFC 3176, September 2001

[RFC-2330] V. Paxson, G. Almes, J. Mahdavi, M. Mathis, Framework for IP Performance Metrics, RFC 2330, May 1998

[RFC-768] J. Postel, "User Datagram Protocol" RFC 768, August 1980

[RFC-3917] J. Quittek, T. Zseby, B. Claise, S. Zander, Requirements for IP Flow Information Export, RFC 3917, October 2004.

[RFC-4271] Y. Rekhter, T. Li, S. Hares, "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, January 2006.

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

[IPFIX-ARCH] G. Sadasivan, N. Browlee, B. Claise, J. Quittek, ''Architecture for IP Flow Information Exp'', RFC-XXXX. [currently internet draft draft-ietf-ipfix-architecture-12, work in progress, September 2006]

[RFC-2819] S. Waldbusser, ''Remote Network Monitoring Management Information Base'', RFC 2819, May 2000.

[Zs02] T. Zseby, ``Deployment of Sampling Methods for SLA Validation with Non-Intrusive Measurements'', Proceedings of Passive and Active Measurement Workshop (PAM 2002), Fort Collins, CO, USA, March 25-26, 2002

15. Authors' Addresses

Derek Chiou
Department of Electrical and Computer Engineering
University of Texas at Austin
1 University Station, Stop C0803, ENS Building room 135,
Austin TX, 78712, USA
Phone: +1 512 232 7722
Email: Derek@ece.utexas.edu

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

Nick Duffield
AT&T Labs - Research
Room B139
180 Park Ave

Florham Park NJ 07932, USA
Phone: +1 973-360-8726
Email: duffield@research.att.com

Albert Greenberg
One Microsoft Way
Redmond, WA 98052-6399
USA
Phone: +1 425-722-8870
Email: albert@microsoft.com

Matthias Grossglauser
School of Computer and Communication Sciences
EPFL
1015 Lausanne
Switzerland
Email: matthias.grossglauser@epfl.ch

Jennifer Rexford
Department of Computer Science
Princeton University
35 Olden Street
Princeton, NJ 08540-5233, USA
Phone: +1 609-258-5182
Email: jrex@cs.princeton.edu

16. Contributors

Sharon Goldberg contributed to Section 12.3 on security considerations for hash-based selection.

Sharon Goldberg
Department of Electrical Engineering
Princeton University
F210-K EQuad
Princeton, NJ 08544, USA
Email: goldbe@princeton.edu

17. Acknowledgements

The authors would like to thank Peram Marimuthu and Ganesh Sadasivan for their input in early versions of this document.

18. Intellectual Property Statements

be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be

available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.
Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

19. Copyright Statement

20. Disclaimer