Internet Draft
Document: <draft-psamp-framework-04.txt>
Expires: April 2004

Nick Duffield (Editor)
AT&T Labs 🔬 Research

October 2003

A Framework for Packet Selection and Reporting

Status of this Memo

Abstract

A wide range of traffic engineering and troubleshooting tasks rely on timely and detailed traffic measurements that can be consistently interpreted. This document describes a framework for packet sampling that is (a) general enough to serve as the basis for a wide range of operational tasks, and (b) needs only a small set of packet selectors that facilitate ubiquitous deployment in router interfaces or dedicated measurement devices, even at very high speeds. The framework also covers reporting and exporting functions used by the sampling host, and configuration of the sampling PSAMP functions.

Comments on this document should be addressed to the PSAMP Working Group mailing list: psamp@ops.ietf.org

To subscribe: psamp-request@ops.ietf.org, in body: subscribe
Archive: https://ops.ietf.org/lists/psamp/

Table of Contents

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.

1. Motivation

This document describes a framework in which to define a standard set of capabilities for network elements to select subsets of packets by statistical and other methods. The framework accommodates ongoing work to (i) specify a set of selectors by which packets are sampled; (ii) specify the information that is to be made available for reporting on sampled packets; (iii) describe a protocol by which information on sampled packets is reported to applications; (iv) describe a protocol by which packet selection and reporting are configured.

The motivation to standardize these capabilities 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, the manner in which the resulting measurements are presented, and consequently, consistency of the interpretation that can be put on them.

The capabilities are positioned as suppliers of packet samples to higher level consumers, including both remote collectors and

applications, and on board measurement-based applications. Indeed, development of the standards within the framework described here should be open to influence by the requirements of standards in related IETF Working Groups, for example, IP Performance Metrics (IPPM) [RFC2330] and Internet Traffic Engineering (TEWG) [LCTV02]. Conversely, we expect that aspects of this framework not specifically concerned with the central issue of packet selection and report formation may be able to leverage work in other Working Groups. Potential examples are the format and export of reports on selected packets, which may leverage the information model and export protocols of IP Flow Information Export (IPFIX) [QZCZ03], and work in congestion aware unreliable transport in the Datagram Congestion Control Protocol (DCCP) [FHK02], and related work in The Stream Control Transmission Protocol (SCTP) [SCTP] and [PR-SCTP].

2. Elements, Terminology and Architecture

This section defines the basic elements of the PSAMP framework. At the highest level, the architecture comprises observation points (at which packets are observed), measurement processes (which select packets and construct reports on them) and export processes (which export reports to collectors). The full definitions of these terms now follow.

* Observation Point: a location in the network where a packet stream is 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 set of interfaces (physical or logical) of a router;

  - an embedded measurement subsystem within an interface.

* Observed Packet Stream: the set of all packets observed at the observation point.

* Packet Stream: either the observed packet stream, or a subset of it.

  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 [QuZC02].

* Selection Process: takes a packet stream as its input and selects a subset of that stream as its output.

* Packet Content: the union of the packet header (which includes link layer, network layer and other encapsulation headers) and

the packet payload.

* Selection State: a selection process may maintain state information for use by the selection process and/or the reporting process. At a given time, the selection state may depend on packets observed at and before that time, and other variables. Examples include:

  - sequence numbers of packets at the input of selectors;

  - a timestamp of observation of the packet at the observation points;

  - iterators for pseudorandom number generators;

- hash values calculated during selection;

- 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.

* Selector: defines the action of a selection process on a single packet of its input. A selected packet becomes an element of the output packet stream of the selection process.

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

  - the packet's content;

  - information derived from the packet's treatment at the observation point;

  - any selection state that may be maintained by the selection process.

* Composite Selection Process: an ordered composition of selection processes, in which the output stream issuing from one component forms the input stream for the succeeding component.

* Primitive Selection Process: a selection process that is not a composite selection process.

* Composite Selector: the selector of a composite selection process.

* Primitive Selector: the selector of a primitive selection process.

* Reporting Process: creates a report stream on packets selected by a selection process, in preparation for export. The input to the

reporting process comprises that information available to the selection process per selected packet, specifically:

- the selected packet's content;

- information derived from the selected packet's treatment at

the observation point;

- any selection state maintained by the inputting selection process, reflecting any modifications to the selection state made during selection of the packet.

* Report Stream: the output of a reporting process is a report stream, comprising two distinguished types of information: packet reports, and report interpretation.

* Packet Reports: a configurable subset of the per packet input to the reporting process.

* Report Interpretation: subsidiary information relating to one or more packets, that is used for interpretation of their packet reports. Examples include configuration parameters of the selection process and of the reporting process.

* Measurement Process: the composition of a selection process that takes the observed packet stream as its input, followed by a reporting process.

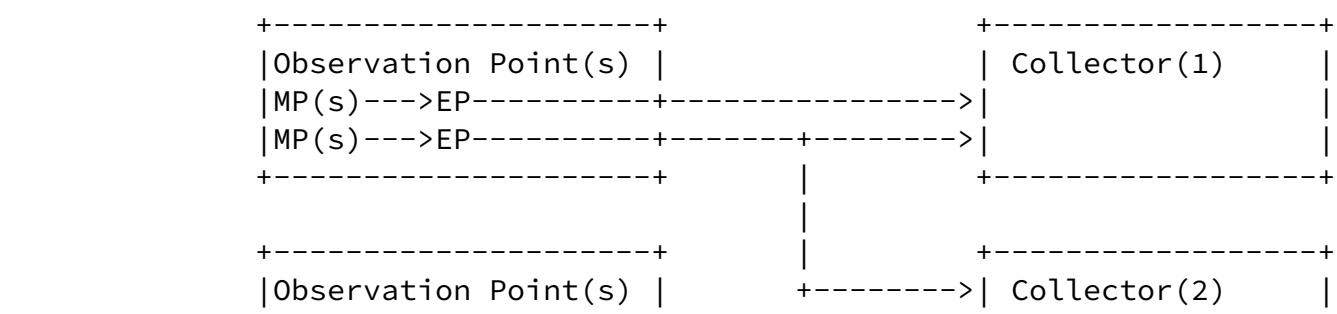
* Export Process: sends the output of one or more reporting processes to one or more collectors.

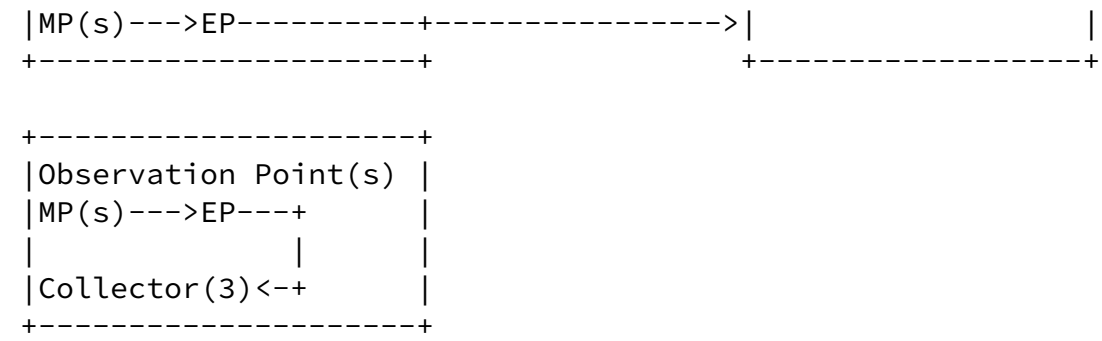
* Collector: a collector receives a report stream exported by one or more export processes. In some cases, the host of the measurement and/or export processes may also serve as the collector.

* Export packets: one or packet reports, and perhaps report interpretation, are bundled by the export process into a export packet for export to a collector.

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

MP = Measurement Process, EP = Export Process





The PSAMP measurement process can be viewed as analogous to the IPFIX metering process. The PSAMP measurement process takes an observed packet stream as its input, and produces packet reports as its output. The IPFIX metering process produces flow records as its output. The distinct name "measurement process" has been retained in order to avoid potential confusion in settings where IPFIX and PSAMP coexist, and in order to avoid the implicit requirement that the PSAMP version satisfy the requirements of an IPFIX metering process (at least while these are under development). The relation between PSAMP and IPFIX is further discussed in [QC03].

3. Requirements

3.1 Selection Process Requirements.

* Ubiquity: The selectors must be simple enough to be implemented ubiquitously at maximal line rate.

* 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.

* Extensibility: to allow for additional packet selectors to support future applications.

* Flexibility: to support selection of packets using different network protocols or encapsulation layers (e.g. IPv4, IPv6, MPLS, etc).

* Robust Selection: packet selection MUST be robust with respect to attempts to craft an observed packet stream from which packets are selected disproportionately (e.g. to evade selection, or overload measurement systems).

* Parallel Measurement Processes: multiple independent measurement

processes at the same host, able to operate simultaneously.

* Non-contingency: in order to satisfy the ubiquity requirement, the selection decision for each packet MUST NOT depend on future packets. Rather, the selection decision MUST be capable of being

made on the basis of the selection process input up to and including the packet in question. This excludes selection functions that require caching of packet for selection contingent on subsequent packets. See also the timeliness requirement following.

Selectors are outlined in Section 4, and described in more detail in the companion document [ZMRD03].

3.2 Reporting Process Requirements

* Transparency: allow transparent interpretation of the report stream, without any need to obtain additional information concerning the observed packet stream.

* Robustness to Information Loss: allow robust interpretation of the report stream with respect to packet reports missing due to data loss, e.g. in transport, or within the selection, reporting or exporting processes. Inclusion in reporting of information that enables the accuracy of measurements to be determined.

* 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 measurement process.

* Privacy: selection of the content of packet reports will be cognizant of privacy and anonymity issues while being responsive to the needs of measurement applications, and in accordance with RFC 2804 [RFC2804]. Full packet capture of arbitrary packet streams is explicitly out of scope.

A specific reporting processes meeting these requirements, and th e requirement for ubiquity, is described in Section 5.

3.3 Export Process Requirements

* Timeliness: packet reports SHOULD be made available to the collector quickly enough to support near real time applications.

Specifically, any report on a packet SHOULD be dispatched within 1 second of the time of receipt of the packet by the measurement process. See Section 5.5 for further discussion of this point

* Congestion Avoidance: export of a report stream across a network MUST be congestion avoiding in compliance with RFC 2914 [RFC 2914].

* Secure Export:

  - confidentiality: the option to encrypt exported data MUST be provided. [MUST vs. SHOULD needs further WG discussion].

  - integrity: alterations in transit to exported data MUST be detectable at the collector

  - 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 [QZCZ03].

3.4 Configuration Requirements

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

* 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 (for example) evade subversion, or overload the measurement infrastructure.

Configuration is discussed in Section 8. Feasibility and complexity of PSAMP operations is discussed in Section 9.

Reuse of existing protocols will be encouraged provided the protocol capabilities are compatible with the requirements laid out in this document.

4. Packet Selection

4.1 Packet Selection Terminology.

* Filtering: a filter is a selection operation that selects a packet deterministically based on the packet content, its treatment, and functions of these occurring in the selection state. Examples include mask/match filtering, and hash-based selection.

* Sampling: a selection operation 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.

* Content-independent Sampling: a sampling operation that does not use packet content (or quantities derived from it) as the basis for selection is called a content-independent sampling operation. 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.

* Content-dependent Sampling: a sampling operation where selection is dependent on packet content is called a content-dependent sampling operation. Examples include pseudorandom selection according to a probability that depends on the contents of a packet field; note that this is not a filter.

* Hash Domain: a subset of the packet content and the packet treatment, viewed as an N-bit string for some positive integer N.

* Hash Range: a set of M-bit strings for some positive integer M.

* Hash Function: a deterministic map from the hash domain into the hash range.

* Hash Selection Range: a subset of the hash range. The packet is selected if the action of the hash function on the hash domain for the packet yields a result in the hash selection range.

* Hash-based Selection: filtering specified by a hash domain, a hash function, and hash range and a hash selection range.

* Approximative Selection: selection operations in any of the above categories may be approximated by operations in the same or another category for the purposes of implementation. For example, uniform pseudorandom sampling may be approximated by hash-based selection, using a suitable hash function and hash domain. In this case, the closeness of the approximation depends on the choice of hash function and hash domain.

* Population size: the number of packets in a subset of a packet stream.

* Sample size: the number of packets selected from a subset of a packet stream by a selection operation.

* Attained Selection Frequency: the actual frequency with which packets are selected by a selection process. The attained sampling frequency is calculated as ratio of the size of a sample size to the size of the population from which it was selected.

* Target Selection Frequency: the long-term frequency with which packets are expected to be selected, based on selector parameter settings. Depending on the selector, the target selection frequency may be count-based or time-based.

  For sampling operations, due to the inherent statistical variability of sampling decisions, the target and attained selection frequencies will not in general be equal, although they may be close in some circumstances, e.g., when the population size is large. In hash-based selection, the target selection frequency is the quotient of size of the hash selection range by the size of the hash range.

4.2 PSAMP Packet Selection Operations

A spectrum of packet selection operations is described in detail in [ZMRD03]. Here we only briefly summarize the meanings for completeness.

A PSAMP selection process MUST support at least one of the following selectors.

* Systematic Time Based Sampling: 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.

* Systematic Count Based Sampling: similar to systematic time based expect that selection is reckoned with respect to packet count rather than time. Packet selection is triggered periodically by packet count, a number of successive packets being selected subsequent to each trigger.

* 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.

* Probabilistic n-out-of-N Sampling: form each count-based successive block of N packets, n are selected at random

* Mask/match Filtering: this entails taking the masking portions of the packet (i.e. taking the bitwise AND with a binary mask) and selecting the packet if the result falls in a range specified in the selection parameters of the filter. This specification doesn't preclude the future definition of a high level syntax for defining filtering in a concise way (e.g. TCP port taking a particular value) providing that syntax can be compiled into the bitwise expression.

  Mask/match operations SHOULD be available for different protocol portions of the packet header:

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

  - transport header

  - encapsulation headers (including MPLS label stack, ATOM, if present)

  When the host of a selection process offers mask/match filtering, and, in its usual capacity other than in performing PSAMP functions, identifies or processes information from one or more of the above protocols, then the information SHOULD be made

available for filtering. For example, when a host routes based on destination IP address, that field should be made available for filtering. Conversely, a host 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.

* Hash-based Selection: Hash-based selection will employ one or more hash functions to be standardized. The hash domain is specified by a bitmaps on the IP packet header and the IP payload.

  When the hash function is sufficiently good, hash-based selection can be used to approximate uniform random sampling over the hash domain. The target sampling frequency is then the ratio of the size of the selection range to the hash range.

  Applications of hash-based selection include:

  - Trajectory Sampling: all routers use the same hash selector; the hash domain includes only portions of the packet that do not change from hop to hop. (For example, in an IP packet, TTL is excluded.) Hence packets are consistently selected in the sense that they are selected at all routers on their path or none. Reports packets also include a second hash (the label hash) that distinguishes different packets. Reports of a given packet reaching the collector from different routers can be used to reconstruct the path taken by the packet. Trajectory sampling is proposed in [DuGr01]; further description is found in [ZMRD03]; some applications are described in Section 10.

  - Consistent Flow Sampling: the hash domain is a flow key. For a given flow, either all or none of its packets are sampled. This is accomplished without the need to maintain flow state.

  Some applications need to calculate packet hashes for purpose other than selection (e.g. the label hash in trajectory sampling). This can be achieved by placing a calculated hash in the selection state, and setting the selection range to be the whole of the hash range.

* Router State Filtering: the selection process MAY support filtering based on the following conditions, which may be combined with the AND, OR or NOT operators:

  - Ingress interface at which packet arrives equals a specified value
  - Egress interface to which packet is routed to equals a specified value
  - Packet violated Access Control List (ACL) on the router
  - Failed Reverse Path Forwarding (RPF)
  - Failed Resource Reservation (RSVP)

- No route found for the packet
- Origin Autonomous System (AS) equals a specified value or lies within a given range
- Destination AS equals a specified value or lies within a given range

Router architectural considerations may preclude some information concerning the packet treatment, e.g. routing state, being available at line rate for selection of packets. However, if selection not based on routing state has reduced down from line rate, subselection based on routing state may be feasible.

4.3 Input Sequence Numbers for Primitive Selection Processes.

Each instance of a primitive selection process 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 frequency at which packets are selected, 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 measurement or reporting process (e.g. due to resource contention in the host of these processes), or loss of export packets in transmission or collection. See [RFC3176] for further details.

4.4 Composite Selectors

The ability to compose selectors in a selection process SHOULD be provided. The following combinations appear to be most useful for applications:

* 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 depend on the composition of the packet stream.

4.5 Constraints on the Sampling Frequency

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

4.6 Criteria for Choice of Selection Operations

In current practice, sampling has been performed using particular algorithms, including:

* pseudorandom independent sampling with probability 1/N;

* systematic sampling of every Nth packet.

The question arises as to whether both of these should be standardized as distinct selection operations, or whether they can be regarded as different implementations of a single selection operation.

To determine the answer to this question, we need to consider

(a) measured or assumed statistical properties of the packet stream, e.g., one or more of the following:

- contents of different packets are statistically independent

- correlations between contents of different packets decay at a specified rate

- contents of certain fields within the same packet are significantly variable and exhibit small cross correlation

(b) the desired reference sampling model, e.g., one of:

- sample packets with long term probability 1/N

- sample packets independent with probability 1/N

(c) the set of possible alternatives and implementations, e.g., one of:

- pseudorandom independent sampling with probability 1/N

- systematic sampling with period N

- hash-based sampling with target probability 1/N

(d) the tolerance for error in the applications that use the collected packet reports.

We can say that a given alternative from (c) reproduces a reference model (b) for the applications if the results obtained using them are sufficiently accurate in (d) for traffic satisfying an assumed statistical properties in (a). Clearly, application to evaluate methods in (c) requires developing agreement on the relevant properties in (a), (b) and (d).

Example: systematic sampling with period N will not count the occurrence of closely space packets (less than N counts apart) from the same flow. Thus for applications that are concerned with the joint statistics of multiple packets within flows, systematic

sampling may not reproduce the results obtained with random sampling sufficiently accurately.

5. Reporting Process

5.1 Mandatory Contents of Packet Reports (MUST)

The reporting process MUST include the following in each packet report:

(i) the input sequence number(s) of any sampling operation that acted on the packet in the instance of a measurement process of which the reporting process is a component.

The reporting process MUST be able to include the following in each packet report, as a configurable option:

(ii) some number of contiguous bytes from the start of the packet, including the packet header (which includes link layer, network layer and other encapsulation headers) and some subsequent bytes of the packet payload.

Some devices hosting reporting processes may not have the resource capacity or functionality to provide more detailed packet reports that those in (i) and (ii) above. Using this minimum required reporting functionality, the reporting process places the burden of

interpretation on the collector, or on applications that it supplies.

5.2 Extended Packet Reports (MAY)

The reporting process MAY provide for the inclusion in packet reports of the following information, inclusion any or all being configurable as a option.

(iii) fields relating to the following protocols used in the packet, specifically: IPv4, IPV6, transport protocols, MPLS, ATOM. Note that optional reporting of field contents may be used to reduce reporting bandwidth, in which case the option to not report information in (ii) above would be exercised.

(iv) packet treatment, including:

- identifiers for any input and output interfaces of the observation point that were traversed by the packet

- source and destination AS

(v) selection state associated with the packet, including:

- the timestamp of observation of the packet at the observation point

- hashes, where calculated.

5.3 Extended Packet Reports in the Presence of IPFIX

If IPFIX 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 [QZCZ03], with modifications appropriate to reporting on single packets rather than flows.

5.4 Report Interpretation

Information for use in 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.

(iv) identifiers for observation point, measurement process, and export process.

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

Identifiers in (iv) are necessary, e.g., in order to match packet reports to the selection process that selected them. For example, when packet reports due to a sampling operation suffer loss (either during export, or in transit) it may be desirable to reconfigure downwards the sampling rate on the selection process that selected them.

The requirements for robustness and transparency are motivations for including report interpretation in the report stream. Inclusion 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.

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.

5.5 Report Timeliness

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 to under 1 second has other benefits besides limiting buffer requirements. For many applications a 1 second time resolution 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.

A dispatch delay of 1 second in these situations eliminates the need for timestamping by synchronized clocks at observation points devices, 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---avoiding the complexity of buffering and reordering samples. See [DuGeGr02] for an example.

6. Parallel Measurement Processes

Because of the increasing number of distinct measurement applications, with varying requirements, it is desirable to set up parallel measurement processes on given observed packet stream. A device capable of hosting a measurement process SHOULD be able to support more than one independently configurable measurement process simultaneously. Each such measurement process SHOULD have the option of being equipped with its own export process; otherwise the parallel measurement processes MAY share the same export process.

Each of the parallel measurement 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 measurement process; other measurement processes need only record that they selected the packet, e.g., by incrementing a counter. The priority amongst measurement processes under resource contention SHOULD be configurable.

It is not proposed to standardize the number of parallel measurement processes.

7. Export Process

7.1 Collector Destination

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

7.2 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 measurement process serve as the input for the IPFIX protocol, which then forms flow records out of the stream of selected packets. Note that IPFIX being still developed; this is given only as a possible example.

7.3 Reliable vs. Unreliable Transport

The export of the report stream does not require reliable export. On the contrary, retransmission of lost export packets consumes additional network resources and requires maintenance of state by the export process. As such, the export process would have to be able to receive and process acknowledgments, and to store unacknowledged data. Furthermore, the host of the export process may not possess its own network address at which to receive acknowledgments. For example an autonomous embedded measurement subsystem in an interface may simply inject export packets into the interface packet stream, designating the interface address as the source address of the export packets). These requirements would be a significant impediment to having ubiquitous support PSAMP.

Instead, it is proposed that the export process support unreliable export. Sequence numbers on the export packets would indicate when loss has occurred, and the analysis of the surviving report stream can be used to determine the degree of loss. In some sense, packet loss becomes another form of sampling (albeit a less desirable, and less controlled, form of sampling).

7.4 Limiting Delay for Export Packets

The export process may queue 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 at the collector as described in Section 5.4.

7.5 Configurable Export Rate Limit

The export process MUST be able to limit its export rate; otherwise it could overload the network and/or the collector. Note this problem would be exacerbated using reliable transport mode, since any lost packets would be retransmitted, thereby imposing an additional load on the network.

At times, the reporting process may generate new packet reports or report interpretation faster than the allowed export rate. In this situation, the export process MUST discard the excess packet reports rather than transmitting them to the collector. Sequence numbers reported for selector input enable correction for lost packet reports. An additional sequence number for dispatched export packets enables the collector to determine the degree of loss in transmission.

There are two options for a configurable rate limit. First, if the transport protocol has a configurable rate limit, that can be used. The second option is to limit the rate at which export packets are supplied to the transport protocol. A candidate for implementation of rate limiting is the leaky bucket, with tokens corresponding e.g. to bytes or packets.

The export rate limit MUST be configurable per export process. Note that since congestion loss can occur at any link on the export path, it is not sufficient to limit rate simply as a function of the bandwidth of the interface out of which export takes place.

7.6 Congestion-aware Unreliable Transport

Export packets compete for resources with other Internet transfers. Congestion-aware export is important to ensure that the export packets do not overwhelm the capacity of the network or unduly degrade the performance of other applications, while making good use of available bandwidth resources.

Choice of transport for PSAMP has to be made under the following

constraints:

(i) IESG has mandated that all transport in new protocols must be congestion aware

(ii) reliable transport is too onerous for general entities that support PSAMP (see Section 7.3)

(iii) there currently exists no IETF standardized unreliable congestion-aware transport

In the absence of an existing IETF standardized unreliable congestion-aware protocol, PSAMP will provisionally nominate the reliable congestion aware transport protocol TCP as the interim transport protocol for export. From the preceding arguments, TCP is unsatisfactory for final standardization in PSAMP. In the meantime, the PSAMP Working Group will evaluate (at least) the following alternatives for congestion aware unreliable transport, as they become available, with a view to selecting one of them and discarding TCP:

(i) unreliable transport protocols adopted in the future by the IPFIX Working Group,

(ii) the Datagram Congestion Control Protocol (DCCP); currently under development; see [FHK02]

(iii) The Stream Control Transmission Protocol (SCTP) under development [SCTP]. SCTP is by default reliable, but has the capability to operate in unreliable and partially reliable modes [PR-SCTP]. See [D03] for description of its potential use in flow export.

(iv) collector-based rate reconfiguration, described below.

7.7 Collector-based Rate Reconfiguration

Since collector-based rate reconfiguration is a new proposal, this draft will discuss it in some detail.

The collector can detect congestion loss along the path from the exporting device to the collector by observing packet loss,

manifest as gaps in the sequence numbers, or the absence of packets for a period of time. The server can run an appropriate congestion-control algorithm to compute a new export rate limit, then reconfigure the export process with the new rate. This is an attractive alternative to requiring the export process to receive acknowledgment packets. Implementing the congestion control algorithm in the collector has the added advantages of flexibility in adapting the sending rate and the ability to incorporate new congestion-control algorithms as they become available.

7.7.1 Changing the Export Rate and Other Rates

Forcing the export process to discard excess packet reports is an effective control under short term congestion. Alternatively, the selection process could be reconfigured to select fewer packets, or the reporting process could be reconfigured to send smaller reports on each selected packet. This may be a more appropriate reaction to long-term congestion. In some cases, a collector may receive export packets due to more than one export process, and could decide to reduce the export or other rates associated with one export process rather than another, in order to prioritize the export packets.

This type of flexibility is valuable for network operators that collect export packets from multiple locations to drive multiple applications.

7.7.2 Notions of Fairness

In some cases, it may be reasonable to allow the collector to have flexibility in deciding how aggressively to respond to congestion. For example, the host of the export process and the collector may have a very small round-trip time (RTT) relative to other traffic. Conventional TCP-friendly congestion control would allocate a very large share of the bandwidth to the PSAMP export traffic. Instead, the collector could apply an algorithm that reacts more aggressively to congestion to give a larger share of the bandwidth to other traffic (with larger RTTs).

In other cases, the export packets may require a larger share of the bandwidth than other flows. For example, consider a link that carries tens of thousands of flows, including some non TCP-friendly DoS attack traffic. Restricting the PSAMP traffic to a fair share allocation may be too restrictive, and might limit the collection of the data necessary to diagnose the DoS attack which overloads links over which export packets are carried. In order to maintain

report collection during periods of congestion, PSAMP report streams may claim more than a fair share of link bandwidth, provided the number of report streams in competition with fair sharing traffic is limited. The collector could also employ policies that allocate bandwidth in certain proportions amongst different measurement processes.

Note that the ability to control differential bandwidth usage in the manner described in this section may be partially or wholly lost if congestion control is performed by other means purely at the transport level.

7.7.3 Behavior Under Overload and Failure

The congestion control algorithm has to be robust to severe overload or complete loss of connectivity between the host of the export process and the collector, and also to the failure of host of the export process or the collector. For example, in a scenario where the collector is unable to reconfigure the export rate because of loss of reverse (collector to exporting host) connectivity, it is desirable for the exporting host to reduce the export rate autonomously. Similarly, if no export packets reach the collector because of loss of forward connectivity, the collector should not react to this by increasing the export rate. This problem may be solved through periodic heartbeat packets in both directions (i.e., export packets in the forward direction, configuration refresh messages in the reverse direction). This allows each side to detect a loss in connectivity or outright failure and to react appropriately.

8. Configuration and Management

A key requirement for PSAMP is the easy reconfiguration of the parameters of the measurement process: those for selection, packet reports and export. Examples are

(i) support of measurement-based applications that want to drill-down on traffic detail in real-time;

(ii) collector-based rate reconfiguration.

To facilitate reconfiguration and retrieval of parameters, they are to reside in a Management Information Base (MIB). Mandatory

configuration, capabilities and monitoring objects will cover all minimum required (MUST) PSAMP functionality.

Secondary objects will cover the recommended PSAMP functionality (SHOULD), and MUST be provided only when such functionality is offered by a host. Such PSAMP functionality includes configuration of offered selectors, composite selectors, multiple measurement processes, and report format including the choice of fields to be reported. For further details concerning the PSAMP MIB, see [DRC03].

PSAMP requires a uniform mechanism with which to access and configure the MIB. SNMP access MUST be provided by the host of the MIB.

9. 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.

9.1 Feasibility

9.1.1 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.

9.1.2 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.

9.1.3 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].

9.1.4 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 reporting process for high sampling rates; reporting selected fields would save on this bandwidth. Thus there is a trade-off between simplicity and bandwidth limitations.

9.1.5 Export

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.

9.2 Potential Hardware Complexity

We now comment on the complexity of possible hardware implementations. Achieving low constants for performance while minimizing hardware resources is, of course, a challenge, especially at very high clock frequencies. Most of these operations, however, are very basic and their implementations very well understood; in fact, the average 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(lg(M)) stages of logic, where M is the number of bits involved in the comparison. The lg(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(lg(M)) stages of logic operation. Optimized implementations of arithmetic operations are also O(lg(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(lg(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(lg(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. However, identification hashing with MD5 on only selected packets is feasible at a sufficiently low sampling frequency.

10. 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. However, there are two major differences:

- PSAMP aims for ubiquitous deployment of packet measurement, including devices that are not expected to support IPFIX. This offers broader reach for existing applications.

- PSAMP can support new applications through the type of packet selectors that it supports

10.1 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 frequency 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 target sampling frequency, or by the attained sampling frequency (as derived by interface 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 measurement process on the same packet stream from which the subset was reported. The selection process of the new measurement process filters according to the attribute of interest, and composes with sampling if necessary to manage the frequency of packet selection.

10.2 Passive Performance Measurement

Hash-based 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. The label hash is used to match up ingress and egress samples. Rates of loss in transit between ingress and egress are estimated from the proportion of trajectories for which no egress report is received. Note 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.

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

10.3 Troubleshooting

PSAMP 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 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 10.1, 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 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 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.

11. Security Considerations

Security considerations are addressed in:

- Section 3.1: item Robust Selection
- Section 3.3: item Secure Export
- Section 3.4: item Secure Configuration

12. References

[B88] R.T. Braden, A pseudo-machine for packet monitoring and statistics, in Proc ACM SIGCOMM 1988

[ClPB93] K.C. Claffy, G.C. Polyzos, H.-W. Braun, Application of Sampling Methodologies to Network Traffic Characterization, Proceedings of ACM SIGCOMM'93, San Francisco, CA, USA, September 13-17, 1993

[DRC03] T. Dietz, D. Romascanu, B. Claise, Definitions of Managed Objects for Packet Sampling, Internet Draft, draft-ietf-psamp-mib-00.txt, work in progress, June 2003.

[D03] M. Djernaes, Cisco Systems NetFlow Services Export Version 9 Transport, Internet Draft, draft-djernaes-netflow-9-transport-00.txt, work in progress, February 2003

[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.

[RFC2914] S. Floyd, Congestion Control Principles, RFC 2914, September 2000.

[FHK02] S. Floyd, M. Handley. E. Kohler, Problem Statement for DCCP, Internet Draft draft-ietf-dccp-problem-00.txt, work in progress, October 2002.

[RFC2804] IAB and IESG, Network Working Group, IETF Policy on Wiretapping, RFC 2804, May 2000

[LCTV02] W.S. Lai, B.Christian, R.W. Tibbs, S. Van den Berghe, A Framework for Internet Traffic Engineering Measurement Internet Draft draft-ietf-tewg-measure-05.txt, work in progress, February 2003.

[RFC3176] P. Phaal, S. Panchen, N. McKee, InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks, RFC 3176, September 2001

[RFC2330] V. Paxson, G. Almes, J. Mahdavi, M. Mathis, Framework for IP Performance Metrics, RFC 2330, May 1998

[QC03] J. Quittek, B. Claise, On the Relationship between PSAMP and IPFIX, Internet Draft draft-quittek-psamp-ipfix-01.txt, work in progress, February 2003.

[QZCZ03] J. Quittek, T. Zseby, B. Claise, S. Zander, Requirements for IP Flow Information Export, Internet Draft draft-ietf-ipfix-reqs-10.txt, work in progress, June 2003.

[SPSJTKS01] A. C. Snoeren, C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, S. T. Kent, W. T. Strayer, Hash-Based IP Traceback, Proc. ACM SIGCOMM 2001, San Diego, CA, September 2001.

[RFC2960] Stewart, R. (ed.) "Stream Control Transmission Protocol", RFC 2960, October 2000

[PR-SCTP] Stewart, R, "SCTP Partial Reliability Extension", Internet Draft, draft-stewart-tsvwg-prsctp-01.txt, work in progress, June 2003.

13. Authors' Addresses

Derek Chiou
Avici Systems
101 Billerica Ave
North Billerica, MA 01862
Phone: +1 978-964-2017
Email: dchiou@avici.com

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 B-139
180 Park Ave
Florham Park NJ 07932, USA
Phone: +1 973-360-8726
Email: duffield@research.att.com

Albert Greenberg
AT&T Labs - Research
Room A-161
180 Park Ave
Florham Park NJ 07932, USA
Phone: +1 973-360-8730
Email: albert@research.att.com

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

Peram Marimuthu
Cisco Systems
170, W. Tasman Drive
San Jose, CA 95134
Phone: (408) 527-6314
Email: peram@cisco.com

Jennifer Rexford
AT&T Labs - Research
Room A-169
180 Park Ave
Florham Park NJ 07932, USA
Phone: +1 973-360-8728
Email: jrex@research.att.com

Ganesh Sadasivan
Cisco Systems
170 W. Tasman Drive
San Jose, CA 95134
Phone: (408) 527-0251
Email: gsadasiv@cisco.com

14. Intellectual Property Statement

AT&T Corporation may own intellectual property applicable to this contribution. The IETF has been notified of AT&T's licensing intent for the specification contained in this document. See http://www.ietf.org/ietf/IPR/ATT-GENERAL.txt for AT&T's IPR statement.

15. Full Copyright Statement