

**Overview of Implementation reports relating to IETF work on IP
Performance Measurements (IPPM, [RFC 2678](#) - 2681)
draft-ietf-ippm-implement-02.txt**

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on July 7, 2006.

Copyright Notice

Copyright (C) The Internet Society (2006).

Abstract

This document contains a summary of the implementation reports submitted for RFC's 2678 to 2681 during the second half of 2004.

Table of Contents

1.	Introduction	4
2.	Requirements notation	4
3.	General Information	5
3.1.	BrixWorx	5
3.1.1.	General	5
3.1.2.	Other, related RFC's implemented	6
3.2.	NetWarrior	6
3.3.	Surveyor	7
3.4.	Test Traffic Measurements	7
3.4.1.	General	7
3.4.2.	Other, related RFC's implemented	8
3.5.	WIPM	8
3.5.1.	General	8
3.5.2.	Other, related RFC's implemented	9
3.5.3.	Common features	9
4.	RFC2678 metrics	10
4.1.	BrixWorx	10
4.2.	NetWarrior	10
4.3.	Surveyor	11
4.4.	TTM	11
4.5.	WIPM	11
4.6.	Conclusion	11
5.	RFC2679 metrics	12
5.1.	BrixWorx	12
5.2.	NetWarrior	12
5.3.	Surveyor	13
5.4.	TTM	15
5.5.	WIPM	16
5.6.	Delay of lost packets	16
5.7.	Conclusion	17
6.	RFC2680 metrics	17
6.1.	BrixWorx	17
6.2.	NetWarrior	17
6.3.	Surveyor	18
6.4.	TTM	18
6.5.	WIPM	19
6.6.	Conclusion	20
7.	RFC2681 metrics	20
7.1.	BrixWorx	20
7.2.	NetWarrior	20
7.3.	Surveyor	20
7.4.	TTM	20
7.5.	WIPM	20
7.6.	Conclusion	21
8.	Security Considerations	21
9.	IANA Considerations	21

Uijterwaal

Expires July 7, 2006

[Page 2]

10.	Conclusion	21
11.	Acknowledgements	21
12.	References	22
	Author's Address	23
	Intellectual Property and Copyright Statements	24

1. Introduction

Over the last years, the IPPM Working Group has developed metrics for various quantities that can be measured on the Internet: Connectivity [[RFC2678](#)], One way delay [[RFC2679](#)], One way packet loss [[RFC2680](#)] and round trip delay [[RFC2681](#)]. These metrics are all based on the IPPM Metrics framework [[RFC2330](#)]. All metrics have the state of proposed standard. It is, however, the goal of the working group to move these documents to Internet draft standards. In order to accomplish these, one must have (at least) 2 interoperable implementations of these RFC's.

During 2004, the chairs of the IPPM WG asked the participants to report any implementations of these metrics. Five reports were submitted. This document contains a summary of these reports.

It should be noted that interoperability for metrics is not a well defined term. All implementations send packets to determine the metrics and each packet will be treated differently by the network elements in its path. Thus, two subsequent measurements of the same metric by two implementations will not yield the same result, However, a large number of measurements of the same quantity with two implementations should produce samples that are statistically similar to each other.

Even though the topic has been discussed in the past, there is currently no accepted Internet standard to determine when two sets of measurements are statistically different from each other or not. We therefor only show what has been implemented but do not (in general) present results to show that two implementations measuring along the same path will yield the same result. Some studies have been done though and these are mentioned where appropriate.

The outline of this document is as follows: [Section 3](#) discusses general details of the 5 implementations. It also lists features that are common to all metrics implemented by a specific implementation. [Section 4](#) through [Section 7](#) then discuss the implementation of a particular RFC.

2. Requirements notation

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 [[RFC2119](#)].

3. General Information

This section contains general information on the implementations as well as information that applies to all metrics implemented by an organization.

All information is taken from contributions by the person listed in the contact details below. No attempt has been made to verify its accuracy. The implementations are listed in alphabetical order.

3.1. BrixWorx

3.1.1. General

Parameter	Value
Name of Implementation	BrixWorx Performance Management System
Mnemonic	BrixMon Performance Monitoring System
Organization	BrixWorx
Contact details	Brix Networks, Inc.
Report submitted by:	285 Mill Road, Chelmsford MA 01824, USA
Origin of code	URL: http://www.brixnet.com
Platform	Kaynam Hedayat
	Written from scratch
	Brix Verifier, Brix Verifier agent for Linux or Windows

The Brix system is a highly scalable and distributed system comprised of hardware and software probes (Brix Verifiers) managed by a central management system (BrixWorx). The Brix Verifiers are responsible for performing tests and collection of performance metrics. BrixWorx is responsible for provisioning and management of Brix Verifiers, data storage, data sampling, user applications, and interfacing with third party systems.

Brix Verifiers are available as hardware appliances or software agents for Linux and Windows. The Brix Verifiers provide wire-time hardware timestamping with microsecond accuracy that excludes any host specific uncertainties. Time synchronization is available with NTP, CDMA, or GPS for one-way measurements in both directions during a single test run. The testing can be performed between Verifiers or between Verifiers and other network components.

Sun Solaris or Windows based BrixWorx central management servers

provides a single point of management for all Verifiers in the network. All measurements are collected by BrixWorx and stored in a central database. The data is used for various applications including SLA, reporting, and root cause analysis.

The platform offers flexible testing with multiple configuration parameters including:

- o Src, the IP address of a host
- o Dst, the IP address of a host
- o TOS/DSCP
- o VLAN tag
- o TTL
- o Payload length
- o Random scheduling of tests
- o ICMP, UDP, TCP packet types
- o Periodic and random sampling of tests

3.1.2. Other, related RFC's implemented

Brix has also implemented the RFC's on One-Way Loss Patterns [[RFC3357](#)] and IP Delay Variations [[RFC3393](#)].

3.2. NetWarrior

Parameter	Value
Name of Implementation	Netwarrior
Organization	QosMetrics, SA.
Contact details	Joe Ovanesian VP Engineering, 802 Calle Plano, Camarillo, CA 93012, USA
	Email: joe_ovanesian@qosmetrics.com
	URL: http://www.qosmetrics.com
Report submitted by:	Yves Cognet, yves@qosmetrics.com , August 25, 2004.
Origin of code	Written from scratch
Platform	Linux 2.4.22.

Hardware timestamping engine TSE (tx and rx) with built in GPS and TXC0. This platform is running IPv4 and IPv6, PPPoE with both static, dynamic and NAT'ed addresses. All tests have been done in both the IPv4 and IPv6 environment (except for PPPoE).

3.3. Surveyor

Parameter	Value
Name of Implementation	Surveyor
Organization	Advanced Network & Services, Inc.
Contact details (original)	200 Business Park DR, Armonk, NY 10504, USA
Contact details (current)	Wisconsin Advanced Internet Lab
	Email: pb@cs.wisc.edu
Report submitted by:	Matthew Zekauskas, August 1, 2004, matt@internet2.edu
Origin of code	Written from scratch
Platform	BSDI/OS 3.2+, TrueTime GPS-PC card, with custom device driver

Advanced is essentially defunct as of this writing (30-Jul-2004); The Surveyor code and data from 1997-2002 were donated to the Wisconsin Advanced Internet Lab. For details on the implementation see the INET'99 paper [[inet99](#)].

This platform is IPv4-only; while the code should be IP-address independent, this was never verified, and would likely require some tweaking.

3.4. Test Traffic Measurements

3.4.1. General

Parameter	Value
Name of Implementation	RIPE NCC Test Traffic Measurements
Mnemonic	TTM
Organization	RIPE NCC
Contact details	P.O.Box 10096, 1001 EB Amsterdam, the Netherlands
	Email: ttm@ripe.net
	URL: http://www.ripe.net/ttm
	Phone: +31.20.5354444
Report submitted by:	Henk Uijterwaal, henk@ripe.net, July 27, 2004

Origin of code	Written from scratch, uses NTP [RFC1305]	
	and BPF.	
Platform	FreeBSD (version 2.2.8, 4.5 and higher)	
+-----+		

[3.4.2.](#) Other, related RFC's implemented

RIPE NCC has also implemented the RFC on IP Delay Variations [[RFC3393](#)].

[3.5.](#) WIPM

[3.5.1.](#) General

+-----+		
Parameter	Value	
+-----+		
Name of	World-wide IP Performance Measurement System	
Implementation		
Mnemonic	WIPM	
Organization	AT&T Labs	
Contact	Len Ciavattone (lencia@att.com), Ron Kulper	
details	(rkulper@att.com), Al Morton (acmorton@att.com)	
	and Gomathi Ramachandran (gomathi@att.com).	
Report	Al Morton, December 13, 2004.	
submitted by:		
Origin of code	Written from scratch	
Platform	Production platform is Sun Nextra T4 (2x	
	UltraSPARC III+), 150 MHz, 2GB memory, Solaris 6	
	and 8.	
	Measurement components also run on Intel with	
	Red Hat 8+, Fedora Core 1 & 2, Knoppix 3.3.	
+-----+		

Measurement paths are determined by performing a traceroute at the start time of each measurement stream. The return path is also determined from traceroute from destination to source.

Measurement servers are located in major network nodes. Results are combined at a single server dedicated to collection, data-feeds, reporting, display, and alarm generation. All servers use two stratum 1 NTP servers in AT&T's network for time-of-day synchronization at the time of this writing. A subset of the results are published continuously at <http://www.att.com/ipnetwork>.

WIPM measures the characteristics of both the Src-Dst and Dst-Src paths in the same test interval by implementing a responder function at the Destination. Today, there is very limited reporting of 1-way

measurements due to time-of-day accuracy limitations, making round-trip delay and loss measurements the most reliable from an accuracy perspective.

WIPM results are stored in Daytona database format, in addition to test-specific files with summary and per-packet (singleton) results.

For further details on the WIPM deployment and measurements see [\[cia03\]](#).

[3.5.2.](#) Other, related RFC's implemented

AT&T Labs has also implemented the RFC on Network measurements with periodic streams [\[RFC3432\]](#), portions of the RFC on IP Loss Patterns [\[RFC3357\]](#) and portions of the RFC on IP Packet Delay Variation [\[RFC3393\]](#).

[3.5.3.](#) Common features

For all metrics in WIMP, the following parameters are implemented:

- o Src, the IP address of a host
- o Dst, the IP address of a host
- o T, a time (singleton)
- o dT, or Th, a time threshold (for declaring loss)
- o T0, a time (stream start)
- o Tf, a time (stream finish)
- o lambda, a rate in reciprocal seconds

The following parameters implemented for [RFC 3432](#) [\[RFC3432\]](#) periodic sampling are:

- o T, the beginning of a time interval where a periodic sample is desired.
- o dT, the duration of the interval for allowed sample start times.
- o T0, a time that MUST be selected at random from the interval [T, T+dT] to start generating packets and taking measurements.
- o Tf, a time, greater than T0, for stopping generation of packets for a sample (Tf may be relative to T0 if desired).
- o incT, the nominal duration of inter-packet interval, first bit to first bit.

Type P capabilities: UDP packets are used. Source Port numbers are variable. Payload length is variable. TOS field or Diffserv Code Points can be set.

Reporting: The parameters above and the measurement path (traceroute) are reported.

Resolution: The time stamp resolution currently in use is 1 ms.

Duplicate Packets: Ignored, first to arrive is used.

Corrupted/Errored Packets: Treated as lost. They will fail either a hardware, IP or UDP checksum verification, and do not reach the measurement app.

Fragmented Packets: Included only if all fragments arrive.

Payload Padding bits: A bit pattern is used instead of random bits.

Errors and Uncertainties (pertaining to clock accuracy): The NTP loopstats and peerstats are logged.

Tests done on the implementation:

- o Verification of results with passive methods: comparison with Packet Sniffer.
- o Error estimation, clock source: Resolution dominates the error.

[4. RFC2678 metrics](#)

[4.1. BrixWorx](#)

The following metrics have been implemented:

- o Type-P-Instantaneous-Unidirectional-Connectivity
- o Type-P-Instantaneous-Bidirectional-Connectivity
- o Type-P-Interval-Unidirectional-Connectivity
- o Type-P-Interval-Bidirectional-Connectivity

The following metrics has not been implemented due to a lack of market demand:

- o Type-P-Interval-Temporal-Connectivity

[4.2. NetWarrior](#)

Implemented is: Type-P-Instantaneous-Bidirectional-Connectivity.

Parameters recorded:

- o Src, the IP address of a host
- o Dst, the IP address of a host
- o T, a time
- o TTL
- o payload length
- o TOS/DSCP
- o VLAN
- o Units: dT

Methodology: + According to [section 3.1 of RFC 2678](#), time synchronization is done by the TSE card, timestamping is done in

hardware on the TSE card.

Comments: TCP packets are being used over ipv4 and ipv6. The TCP checksum is over written when the frame is sent and when the TSE is appending the time stamp. The port numbers for test traffic can be specified. User data length is user defined. Access to special IP bits (TOS/DSCP values) is available. If a packet is duplicated, any duplicates are counted (the delay of the first packet to arrives is used). Corrupted packets are counted as lost; they will fail either a hardware CRC check, or IP or UDP checksum verification.

Features included: path variation (TTL variation) is being recorded.

Reporting the metric: all results are recorded in an SQL database, good and bad records (lost of synchronization), all timestamps are rpeorted in UTC. No aggregation is done by default, but this can be done locally.

[4.3.](#) Surveyor

Has not implemented this RFC.

[4.4.](#) TTM

Has not implemented this RFC due to a lack of demand from its customers. TTM does measure packet loss and assumes that when packet loss is 100%, there is no connectivity.

[4.5.](#) WIPM

Implemented is: Type-P-Instantaneous-Bidirectional-Connectivity.

At the outset of all WIPM tests, there is a packet exchange between the source and destination hosts (1 packet in each direction). If this exchange is successful, the test will continue. These packets assess Bidirectional Connectivity for A1-A2 at time T, and A2-A1 at time T+dT, and are not included in the sample for any other metrics. The commencement of the test stream is proof of this connectivity (and the availability of test resources, which is seldom at issue).

The remaining metrics in this RFC have not been implemented. Other metrics were deemed more useful.

[4.6.](#) Conclusion

Only the Type-P-Instantaneous-Bidirectional-Connectivity metric has been implemented by 2 vendors.

5. [RFC2679](#) metrics

5.1. BrixWorx

- o Type-P-One-way-Delay: implemented.
- o Type-P-One-Way-Delay-Poisson-Stream: implemented.
- o Type-P-One-Way-Delay-Percentile: Implemented, configurable.
- o Type-P-One-Way-Delay-Median: implemented.
- o Type-P-One-Way-Delay-Minimum: implemented.
- o Type-P-One-Way-Delay-Inverse-Percentile: Not implemented due to lack of interest from customers.

Hardware timestamp provides wire-time. CDMA, GPS or NTP is used for time synchronization.

5.2. NetWarrior

- o Implemented: Type-P-One-way-Delay.
 - * Parameters:
 - + Src, the IP address of a host
 - + Dst, the IP address of a host
 - + T, a time
 - + Payload
 - + VLAN ID
 - + Units: dt
 - * Methodology: According to [section 3.6 of RFC 2679](#), Synchronization via TSE card, Timestamping in hardware via TSE card
 - * Comments: UDP and TCP packets are being used over ipv4 and ipv6. TCP checksum is over written when the frame is sent and when the TSE is appending the time stamp. The port numbers for test traffic can be specified among sessions, although for any given session port numbers were fixed. User data length was 12 bytes (32 bit sequence number, 64 bit time value). Access to special IP bits (TOS/DSCP values) is available. If a packet is duplicated, any duplicates are counted (the delay of the first packet to arrives is used). Corrupted packets are counted as lost; they will fail either a hardware CRC check, or IP or UDP checksum verification.
 - * Features included: If either end lost synchronization (as reported by the TSE timing card) the session is kept but results are marked invalid. Path variation (TTL variation) is recorded.
 - * Reporting the metric: All results are recorded in an SQL database, good and bad records (lost of synchronization). The loss threshold is 1 received packets for UPD traffic. All timestamps are reported in UTC Local aggregation can be done. TTL variation are being recorded.

- * Tests performed: calibration, measured time uncertainty is in the 450 ns range end to end once TSE cards are in sync.
- o Type-P-One-way-Delay-Percentile: By default, the 50th percentile (median) and 90th percentile are calculated and reported for x minute intervals (user setup).
- o Type-P-One-way-Delay-Median: This value can be calculated and reported from the database.
- o Type-P-One-way-Delay-Minimum: This value is reported for x minute intervals (user setup).
- o Type-P-One-way-Delay-Poisson-Stream: Not implemented.
- o [RFC2679](#)/Type-P-One-way-Delay-Inverse-Percentile.

5.3. Surveyor

- o Type-P-One-way-Delay
 - * Parameters:
 - + Src, the IP address of a host.
 - + Dst, the IP address of a host.
 - + T, a time.
 - + Units: dt.
 - * Methodology: According to [section 3.6 of RFC 2679](#). Synchronization via GPS time cards. Timestamping in software (and not in the network driver, although we did have an experimental version that did this).
 - * Comments (Type-P): UDP packets are being used. The port numbers for test traffic varied (and could not be specified) among sessions, although for any given session port numbers were fixed. User data length was 12 bytes (32 bit sequence number, 64 bit time value). In general, no special IP bits are set, but there was a version of the code which allowed the user to set the TOS/DSCP values. If a packet is duplicated, any duplicates are ignored (the delay of the first packet to arrives is used). Corrupted packets are generally counted as lost; they will fail either a hardware CRC check, or IP or UDP checksum verification.
 - * Features included: If either end lost synchronization with GPS (as reported by the timing card) the session was broken and measurement ceased.
 - * Reporting the metric: Because measurements ceased when GPS synchronization was lost, and the errors (typically 50 to 100 microseconds with our densest measurement mesh) were much smaller than instrumented paths, negative values were not reported, but flagged as a possible software error. The loss threshold was 400 received packets in our implementation. For typical paths where two packets per second (on average) were sent, the loss threshold was about 200 seconds, or 3 1/3 minutes. In our implementation the distinguished value 10,000,000 was used to signify infinite delay (a loss). (10

- seconds -- although in practice I do not believe we saw anything greater than ~3 sec.) All systematic errors are removed from the timestamps before reporting the metrics. Paths are being determined by running the well-known traceroute tool approximately 6 times an hour (on a Poisson schedule with average 10 seconds, but also clamped to 10 seconds maximum). Paths were stored independent of delays, but displayed along with delays in our graphical displays.
- * Tests performed: Back-to-back tests of typical machines in the field was performed for calibration. For our worst-case fan-out (due to the software implementation, error became worse the more paths that were measured), 80 paths, the calibration error was 100 microseconds. Cross check of measurements against the RIPE-TTM implementation (reported at the 44th IETF IPPM WG meeting by Henk Uijterwaal).
- o Type-P-One-way-Delay-Poisson-Stream
 - * Parameters:
 - + Src, the IP address of a host.
 - + Dst, the IP address of a host.
 - + T0, a time.
 - + Tf, a time.
 - + lambda, a rate in reciprocal seconds. (packets per second).
 - * Units:
 - + T, a time.
 - + dT, either a real number or an undefined number of seconds.
 - * Report consists of series of measurements for Type-P-One-Way-Delay. All the Type-P-One-Way-Delay features and comments above apply here.
 - * Features: The receiver knew the send schedule (passed as a seed to the random number generated used for the Poisson schedule) so it could independently determine if packets were lost. As with the singleton metrics, the loss threshold was 400 received packets in our implementation. For typical paths where two packets per second (on average) were sent, the loss threshold was about 200 seconds, or 3 1/3 minutes. This 400 packet window was also used to correct for reordering. All delay values were kept to allow computation of arbitrary percentiles on demand.
 - o Type-P-One-way-Delay-Percentile: By default, the 50th percentile (median) and 90th percentile are calculated and reported for one minute intervals. A separate Java-based UI could ask for arbitrary percentiles over arbitrary intervals (provided the raw data was on-line) -- typically 6 months of raw data were on-line.
 - o Type-P-One-way-Delay-Median: This value is calculated and reported for one minute intervals, although we did have a separate Java-based UI that could vary the reporting period.

- o Type-P-One-way-Delay-Minimum: This value is calculated and reported for one minute intervals, although we did have a separate Java-based UI that could vary the reporting period.
- o Type-P-One-way-Delay-Inverse-Percentile: Not implemented. Our implementation did not calculate any inverse percentiles. One could ask for the raw data and calculate the inverse percentile yourself. Our implementation did, however, provide a histogram of delay values, by default ranging from 0 to 300mS in 10mS buckets.

[5.4.](#) TTM

- o Type-P-One-way-Delay
 - * Parameters:
 - + Src, the IP address of a host.
 - + Dst, the IP address of a host.
 - + T, a time.
 - * Units: dT.
 - * Methodology: According to [section 3.6 of RFC 2679](#).
 - * Comments: UDP packets are being used. No special IP bits are set.
 - * Features included: A bit to record if the src thinks its clock is synchronized or not (based on NTP). Measurements with unsynchronized clocks are not used in higher level statistics. A bit to record if the dst thinks its clock is synchronized or not. Experimental error estimate for src clock. Experimental error estimate for dst clock. Experimental error estimate for the measurement. Ports used at source and destination.
 - * Features not implemented: No special IP bits can be set.
 - * Reporting the metric: Negative delay values are reported. Small negative values can occur when the experimental errors on the clocks are large and the delays are small. Maximum delay is 255 seconds, packets with higher delays are considered lost. Packet size is being reported. All systematic errors are removed from the timestamps before reporting the metrics. Paths are being determined by running the well-known traceroute tool approximately 10 times an hour, with the most recent path reported.
 - * Tests performed: Cross check of measurements against the surveyor implementation (Reported at the IETF44, IPPM WG meeting). Cross check of measurements with passive measurements (Reported at PAM2001).
- o Type-P-One-way-Delay-Poisson-Stream
 - * Parameters:
 - + Src, the IP address of a host.
 - + Dst, the IP address of a host.
 - + T0, a time.

- + T_f , a time.
- + λ , a rate in reciprocal seconds, reported as packets per minute.
- * Units:
 - + T , a time.
 - + dT , either a real number or an undefined number of seconds.
- * Report consists of series of measurements for Type-P-One-Way-Delay.
- o Type-P-One-way-Delay-Median: This value is being calculated and reported, for 15 minutes (or longer) intervals. 15 minutes came from user feedback.
- o Type-P-One-way-Delay-Minimum: Not implemented. We do, however, report the 2.5% and 97.5% of the distribution for 15 minutes (or longer) intervals. All individual measurements are available to calculate arbitrary percentiles on request.

5.5. WIPM

- o Type-P-One-way-Delay (singleton). Metric Units: dT and T are recorded. The delay of lost packets is undefined, so our implementation of this metric is more accurately described as Type-P-Finite-One-way-Delay. Methodology is as described in [Section 3.6](#).
- o Type-P-One-way-Delay-Poisson-Stream. Metric Units: dT and T are recorded. Methodology is as described in [section 4.6](#), including the correct handling of out-of-order packets.
- o Type-P-One-way-Delay-Periodic-Stream: implemented, see previous item.
- o Type-P-(Finite-)One-way-Delay-Percentile: Computation and report of 0.1, 1.0, 50, 95, 99, and 99.9 percentiles.
- o Type-P-(Finite-)One-way-Delay-Median: implemented.
- o Type-P-(Finite-)One-way-Delay-Minimum: Computation and report of minimum delay (dT) of the sample.
- o Type-P-(Finite-)One-way-Delay-Inverse-Percentile: The fraction of finite dT values greater than a threshold are reported.

Also reported in summary file: Mean, Standard Deviation, and Sum.

Metrics and features not implemented, and why not: We do not treat lost packets as having infinite delay, as it would prevent the calculation of Means. This approach is consistent with ITU-T Rec. Y.1540.

5.6. Delay of lost packets

The RFC specifies that packets that are lost during transmission, are assigned a value for the delay of "infinite", and should be included when calculating statistics over a sample of packets (such as, for

example, the median delay).

This approach suffers from two problems: First, infinite is not a number and thus prevents one from calculating the average delay or any other parameter involving the values of sample of packets. Infinite is also not a number that can be assigned to a variable in most programming languages. Second, including lost packets when reporting the metric, essentially results in report of a metric that combines two parameters (loss and delay), while the user is usually interested in values for the individual metrics.

To circumvent this problem, all implementations assign a special but numeric value to the delay of lost packets. (For example, -1.0 seconds). Lost packets are excluded when calculating statistics over the sample of packets.

[5.7.](#) Conclusion

The basic metric (Type-P-One-way-Delay) has been implemented by all implementations. The statistics that are reported vary from implementation to implementation and depend on the feedback from the users of a particular product. However, as the raw data is available, one can always calculate the missing parameters after the measurement. When advancing this, the paragraph on delays of lost packets should be revisited and possibly reworded.

[6.](#) [RFC2680](#) metrics

[6.1.](#) BrixWorx

- o Type-P-One-way-Packet-Loss: implemented.
- o Type-P-One-way-Packet-Poisson-Stream: implemented.
- o Type-P-One-way-Packet-Loss-Average: implemented.

GPS, CDMA or NTP is used for time synchronization, time-outs are configurable.

[6.2.](#) NetWarrior

- o Type-P-One-way-Packet-Loss.
 - * Metric Parameters:
 - + Src, the IP address of a host.
 - + Dst, the IP address of a host.
 - + T, a time.
 - * Comments: The time is the time the packet was sent, 40 nano second ticks. Packets that arrive at the machine corrupted are dropped (CRC failure or by IP or UDP checksum failure). If a

- packet is duplicated, any duplicates are ignored.
- o Type-P-One-way-Packet-Loss-Poisson-Stream: Not implemented.
- o Type-P-One-way-Packet-Loss-Average: Not implemented.

6.3. Surveyor

- o Type-P-One-way-Packet-Loss
 - * Metric Parameters:
 - + Src, the IP address of a host.
 - + Dst, the IP address of a host.
 - + T, a time.
 - * Metric Units: Type-P-One-way-Packet-Loss = [0,1]
 - * Comments: The time is the time the packet was sent, rounded to the nearest microsecond. Packets that arrive at the machine corrupted are generally dropped either by the hardware CRC failure, or by IP or UDP checksum failure. Hence, corrupted packets are counted as lost. If a packet is duplicated, any duplicates are ignored.
 - * Further comments: This metric has been implemented in combination with [RFC2679](#)/Type-P-One-way-Delay. All comments made there apply. If a packet arrives, it's Type-P-One-way-Packet-Loss is 0. If a packet does not arrive in the specified window (200 seconds by default) the delay is infinite (undefined), and the Type-P-One-way-Packet-Loss is 1.
- o Type-P-One-way-Packet-Loss-Poisson-Stream: This has been implemented as part of [RFC2679](#)/Type-P-OWD-Poisson-Stream. Results consists of a series of measurements for the One Way Delay, with values of 10,000,000 for packets that were sent but did not arrive.
- o Type-P-One-way-Packet-Loss-Average: By default, the Surveyor reporting mechanisms calculate this on one-minute intervals. A Java UI allows calculations on different intervals.

6.4. TTM

- o Type-P-One-way-Packet-Loss.
 - * Metric Parameters:
 - + Src, the IP address of a host.
 - + Dst, the IP address of a host.
 - + T, a time.
 - * Metric Units: Type-P-One-way-Packet-Loss.
 - * Comments: UDP packets are being used. No special IP bits are set. A packet is considered lost if it has not arrived after 255 seconds. There is no distinction between packets that do arrive but take longer than 255 seconds, or that never arrive at all. The time is the time the packet was sent, rounded to the nearest full second.

- * Further comments: This metric has been implemented in combination with [RFC2679](#)/Type-P-OWD. All comments made there apply.
- o Type-P-One-way-Packet-Loss-Poisson-Stream: This has been implemented as part of [RFC2679](#)/Type-P-OWD-Poisson-Stream. Results consists of a series of measurements for the One Way Delay, with values of -1 for packets that were sent but did not arrive.
- o Type-P-One-way-Packet-Loss-Average: This is being calculated for 15 minute (or longer) intervals.

[6.5](#). WIPM

- o Type-P-One-way-Packet-Loss.
 - * Metric Units: Successful transmission and Loss are indicated, but not with the values 0 and 1.
 - * In general, the methodology of [section 2.6](#) is followed. All packets are allowed at least $T_h=3$ seconds to arrive (see the Stream section).
 - * Errors and Uncertainties: Measurement system "health metrics" are periodically checked to ensure that resource limits are not exceeded.
- o Type-P-One-way-Loss-Periodic-Stream
 - * Metric Units: T and a finite delay are recorded for successful packets. At present, T is not recorded for lost packets, though it can be bounded using the sequence numbers applied to successful packets. T will be reported for lost packets in a future release.
 - * Methodology is as described in [section 3.6](#), including the correct handling of out-of-order packets. The threshold for lost packets is $T_h=3$ seconds for the LAST packet in the stream. In other words, a sample with $T_f-T_0=900$ seconds allows $T_h=903$ seconds for the first packet in the sample. A constant T_h setting can be enforced by post-processing the finite delays. T_h is enforced on the Src-Dst-Src RTT in the WIPM responder architecture.
- o Type-P-One-way-Loss-Poisson-Stream: Implemented, see above.
- o Type-P-One-way-Loss-Average: This and many [RFC 3357](#) Loss patterns metrics are reported.

Metrics and features not implemented, and why not: The degree of synchronization between source and destination clocks can be derived with limited accuracy, but it is not directly reported.

The Anderson-Darling test results for the Poisson Process are not available at the time of this memo, however other tests assure the similarity of our Poisson implementation to ideal.

[6.6.](#) Conclusion

As was the case for delay, the basic metric Type-P-One-way-Packet-Loss has been implemented by all implementations. Which statistics are reported varies but, again, all statistics can be calculated offline after the measurement.

[7.](#) [RFC2681](#) metrics

[7.1.](#) BrixWorx

Implemented metrics are:

- o Singleton: Type-P-Round-Trip-Delay.
- o Sample: Type-P-Round-Trip-Delay-Poisson-Stream.
- o Statistical: Type-P-Round-Trip-Delay-Percentile, configurable.
- o Statistical: Type-P-Round-Trip-Delay-Median.
- o Statistical: Type-P-Round-Trip-Delay-Minimum.

Not implemented is:

- o Statistical: Type-P-Round-Trip-Delay-Inverse-Percentile, no market demand.

[7.2.](#) NetWarrior

Has not implemented this RFC, if round trip results are needed, they are created by adding one-way delays.

[7.3.](#) Surveyor

Has not implemented this RFC.

[7.4.](#) TTM

Has not implemented this RFC due to a lack of demand from its customers

[7.5.](#) WIPM

Implemented metrics are:

- o Type-P-Round-trip-Delay. Metric Units: T and Round trip time are recorded. The delay of lost packets is undefined, so our implementation of this metric is more accurately described as Type-P-Finite-Round-trip-Delay. Methodology is as described in [Section 2.6](#), in general. If the packet return transmission is delayed at the destination, the processing time is recorded and inserted in the packet.

- o Type-P-Round-trip-Delay-Poisson-Stream and
 - o Type-P-Round-trip-Delay-Periodic-Stream. Metric Units: T and Round trip Time, dT, are recorded. Methodology is as described in [Section 3.6](#), in general including the correct handling of out-of-order packets.
 - o Type-P-(Finite-)Round-trip-Delay-Percentile and
 - o Type-P-(Finite-)Round-trip-Delay-Median. Computation and report of the 0.1, 1.0, 50, 95, 99, and 99.9 percentiles.
 - o Type-P-(Finite-)Round-trip-Delay-Minimum: Computation and report of minimum round-trip delay (dT) of the sample.
 - o Type-P-(Finite-)Round-trip-Delay-Inverse-Percentile: The fraction of finite dT values greater than a threshold are reported.
- Also calculated and reported are: Mean, Standard Deviation, and Sum.

Note: We do not treat lost packets as having infinite delay, as it would prevent the calculation of Means.

[7.6.](#) Conclusion

This metric has been implemented by 2 vendors.

[8.](#) Security Considerations

All security concerns raised in the specification of the various metrics apply to this report.

[9.](#) IANA Considerations

None.

[10.](#) Conclusion

All metrics have been implemented by multiple vendors and can be moved to draft standard.

[11.](#) Acknowledgements

The authors would like to thank all implementors who submitted an implementation report: Yves Cognet (QosMetrics) for Netwarrior, Kaynam Hedayat (Brix) for BrixWorx, Al Morton (AT&T) for WIPM, Henk Uijterwaal (RIPE NCC) for TTM and Matt Zekauskas (Internet2) for Surveyor.

12. References

- [RFC1305] Mills, D., "Network Time Protocol (Version 3) Specification, Implementation", [RFC 1305](#), March 1992.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2330] Paxson, V., Almes, G., Mahdavi, J., and M. Mathis, "Framework for IP Performance Metrics", [RFC 2330](#), May 1998.
- [RFC2678] Mahdavi, J. and V. Paxson, "IPPM Metrics for Measuring Connectivity", [RFC 2678](#), September 1999.
- [RFC2679] Almes, G., Kalidindi, S., and M. Zekauskas, "A One-way Delay Metric for IPPM", [RFC 2679](#), September 1999.
- [RFC2680] Almes, G., Kalidindi, S., and M. Zekauskas, "A One-way Packet Loss Metric for IPPM", [RFC 2680](#), September 1999.
- [RFC2681] Almes, G., Kalidindi, S., and M. Zekauskas, "A Round-trip Delay Metric for IPPM", [RFC 2681](#), September 1999.
- [RFC3357] Koodli, R. and R. Ravikanth, "One-way Loss Pattern Sample Metrics", [RFC 3357](#), August 2002.
- [RFC3393] Demichelis, C. and P. Chimento, "IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)", [RFC 3393](#), November 2002.
- [RFC3432] Raisanen, V., Grotefeld, G., and A. Morton, "Network performance measurement with periodic streams", [RFC 3432](#), November 2002.
- [cia03] L. Ciavattone, A. Morton and G. Ramachandran, ""Standardized Active Measurements on a Tier 1 IP Backbone", IEEE Communications Magazine, pp 90-97", July 2003.
- [inet99] Sunil Kalidindi and Matthew Zekauskas, "Surveyor: An Infrastructure for Internet Performance Measurements, in proceedings of INET'99", 1999.

Author's Address

Henk Uijterwaal
RIPE NCC
Singel 258
1016 AB Amsterdam
NL

Phone: +31 20 535 4444
Email: henk@ripe.net

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might 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.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2006). This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.

