Network Working Group
Request for Comments: 3763
Category: Informational

S. Shalunov
B. Teitelbaum
Internet2

April 2004

One-way Active Measurement Protocol (OWAMP) Requirements

Status of this Memo

Copyright Notice

Abstract

With growing availability of good time sources to network nodes, it becomes increasingly possible to measure one-way IP performance metrics with high precision. To do so in an interoperable manner, a common protocol for such measurements is required. This document specifies requirements for a one-way active measurement protocol (OWAMP) standard. The protocol can measure one-way delay, as well as other unidirectional characteristics, such as one-way loss.

1. Motivations and Goals

The IETF IP Performance Metrics (IPPM) working group has proposed standards track metrics for one-way packet delay [RFC2679] and loss [RFC 2680] across Internet paths. Although there are now several measurement platforms that implement the collection of these metrics ([CQOS], [BRIX], [RIPE], [SURVEYOR]), there is not currently a standard for interoperability. This requirements document is aimed at defining a protocol that allows users to do measurements using devices from different vendors at both ends and get meaningful results.

With the increasingly wide availability of affordable global positioning system (GPS) and CDMA based time sources, hosts increasingly have available to them time sources that allow hosts to time-stamp packets with accuracies substantially better than the delays seen on the Internet--either directly or through their proximity to NTP primary (stratum 1) time servers. By standardizing a technique for collecting IPPM one-way active measurements, we hope to create an environment where these metrics may be collected across

a far broader mesh of Internet paths than is currently possible. One particularly compelling vision is of widespread deployment of open one-way active measurement beacons that would make measurements of one-way delay as commonplace as measurements of round-trip time are today using ICMP-based tools like ping. Even without very accurate timestamps one can measure characteristics such as loss with quality acceptable for many practical purposes, e.g., network operations.

To support interoperability between alternative OWAMP implementations and make possible a world where "one-way ping" could become commonplace, a standard is required that specifies how test streams are initiated, how test packets are exchanged, and how test results are retrieved. Detailed functional requirements are given in the subsequent section.

2. Functional Requirements

The protocol(s) should provide the ability to measure, record, and distribute the results of measurements of one-way singleton network characteristics such as characteristics defined in [RFC2679] and [RFC2680]. Result reporting, sampling, and time stamps are to be within the framework of [RFC2330].

It should be possible to measure arbitrary one-way singleton characteristics (e.g., loss, median delay, mean delay, jitter, 90th percentile of delay, etc.); this is achieved by keeping all the raw data for post-processing by the final data consumer, as specified in section 2.1. Since RFC2679 and RFC2680 standardize metrics based on Poisson sampling processes, Poisson streams must be supported by the protocol(s).

Non-singleton characteristics (such as those related to trains of packets, back-to-back tuples, and so forth) and application traffic simulation need not be addressed. However, they may be addressed if considered practical and not in contradiction to other design goals.

2.1. Keeping All Data for Post-processing

To facilitate the broadest possible use of obtained measurement results, the protocol(s) should not necessitate any required post-processing. (This does not apply to implementation details such as converting timestamps from ticks since midnight into a canonical form or applying calibration constants; such details should naturally be hidden.) All data obtained during a measurement session should be available after the session is finished if desired by the data consumer so that various characteristics can be computed from the raw data using arbitrary algorithms.

2.2. Result Distribution

A means to distribute measurement results (between hosts participating in a measurement session and beyond) should be provided. Since there can exist a wide variety of scenarios as to where the final data destination should be, these should be invoked separately from measurement requests (e.g., receiver should not have to automatically send measurement results to the sender, since it may be the receiver or a third host that are the ultimate data destination).

At the same time, ability to transfer results directly to their destination (without need for potentially large intermediate transfers) should be provided.

2.3. Protocol Separation

Since measurement session setup and the actual measurement session (i) are different tasks; (ii) require different levels of functionality, flexibility, and implementation effort; (iii) may need to run over different transport protocols, there should exist two protocols: one for conducting the actual measurement session and another for session setup/teardown/confirmation/retrieval. These protocols are further referred to as OWAMP-Test and OWAMP-Control, respectively.

It should be possible to use devices that only support OWAMP-Test but not OWAMP-Control to conduct measurement sessions (such devices will necessarily need to support one form of session setup protocol or the other, but it doesn’t have to be known to external parties).

OWAMP-Control would thus become a common protocol for different administrative domains, which may or may not use it for session setup internally.

2.4. Test Protocol

The test protocol needs to be implemented on all measurement nodes and should therefore have the following characteristics:

+ Be lightweight and easy to implement.

+ Be suitable for implementation on a wide range of measurement nodes.

+ Allow UDP as the transport protocol, since the protocol needs to be able to measure individual packet delivery times and has to run on various machines (see the section "Support for Measurements with Different Packet Types" below for further discussion).

+ Support varying packet sizes and network services (e.g., DSCP marking).

+ Be as simple as possible, but no simpler than necessary to implement requirements set forth in this document; the OWAMP-Test packet format should include only universally meaningful fields, and minimum number of them.

+ If practical, it should be possible to generate OWAMP-Test packets small enough, so that when encapsulated, each fits inside a single ATM cell.

+ Data needed to calculate experimental errors on the final result should be included in every OWAMP-Test packet.

2.5. Control Protocol

Control protocol needs to provide the capability to:

+ authenticate peers to each other using a common authentication method that doesn't require building any new authentication infrastructure, such as user ID and a shared secret;

+ schedule zero or more OWAMP-Test sessions (which do not have to be between the peers of OWAMP-Control conversation);

+ start OWAMP-Test sessions simultaneously or at a pre-scheduled per-session times;

+ retrieve OWAMP-Test session results (of OWAMP-Test sessions scheduled in the current and other OWAMP-Control sessions);

+ confirm graceful completion of sessions and allow either side to abort a session prematurely.

The OWAMP-Control design should not preclude the ability to record extended periods of losses. It should always provide peers with the ability to distinguish between network and peer failures.

2.6. Support for Measurements with Different Packet Types

Since the notion of a packet of type P from [RFC2330], section 13 doesn't always imply precise definition of packet type, some decisions narrowing the scope of possible packet types need to be made at measurement protocol design stage. Further, measurement with packets of certain types, while feasible in more closed settings than those implied by OWAMP, become very hard to perform in an open inter-domain fashion (e.g., measurements with particular packets with broken IP checksum or particular loose source routing options).

In addition, very general packet type specification could result in several problems:

+ Many OWAMP-Test speakers will be general purpose computers with a multitasking operating system that includes a socket interface. These will inevitably have higher losses when listening to raw network traffic. Raw sockets will induce higher loss rate than one would have with UDP measurements.

+ It's not at all clear (short of standardizing tcpdump syntax) how to describe formally the filter that a receiver should use to listen for test traffic.

+ Suppose an identity of an authenticated user becomes compromised. Now the attacker could use that to run TCP sessions to the rlogin port of machines around servers that trust this user to perform measurements (or, less drastically, to send spam from that network). The ability to perform measurements is transformed into an ability to generate arbitrary traffic on behalf of all the senders an OWAMP-Control server controls.

+ Carefully crafted packets could cause disruption to some link-layer protocols. Implementors can't know what to disallow (scrambling is different for different link-layer technologies).

It appears that allowing one to ask a measurement server to generate arbitrary packets becomes an unmanageable security hole and a formidable specification and implementation hurdle.

For these reasons, we only require OWAMP to support a small subspace of the whole packet type space. Namely, it should be possible to conduct measurements with a given Differentiated Services Codepoint (DSCP) [RFC2474] or a given Per Hop Behavior Identification Code (PHB ID) [RFC3140].

3. Scalability

While some measurement architecture designs have inherent scalability problems (e.g., a full mesh of always-on measurements among N measurement nodes requires O(N^2) total resources, such as storage space and link capacity), OWAMP itself should not exaggerate the problem or make it impossible (where it is in principle possible) to design other architectures that are free of scalability deficiencies.

It is the protocol user's responsibility to decide how to use the protocol and which measurements to conduct.

4. Security Considerations

4.1. Authentication

It should be possible to authenticate peers to each other using a user ID and a shared secret. It should be infeasible for any external party without knowledge of the shared secret to obtain any information about it by observing, initiating, or modifying protocol transactions.

It should also be infeasible for such party to use any information obtained by observing, modifying or initiating protocol transactions to impersonate (other) valid users.

4.2. Authorization

Authorization shall normally be performed on the basis of the authenticated identity (such as username) and the specification shall require all implementations to support such a mode of authorization. Different identities (or classes of identities) can have different testing privileges. The use of authorization for arriving at specific policy decisions (such as whether to allow a specific test with a specific source and destination and with a given test send schedule -- which would determine the average network capacity utilization -- at a given time) is up to the users.

4.3. Being Hard to Interfere with by Applying Special Treatment to Measurement Packets

The design of the protocol should make it possible to run sessions that would make it very difficult for any intermediate party to make results appear better than they would be if no interference was attempted.

This is different from cryptographic assurance of data integrity, because one can manipulate the results without changing any data in the packets. For example, if OWAMP-Test packets are easy to identify (e.g., they all come to a well-known port number), an intermediate party might place OWAMP-Test traffic into a priority queue at a congested link thus ensuring that the results of the measurement appear better than what would be experienced by other traffic. It should not be easy for intermediate parties to identify OWAMP-Test packets (just as it should not be easy for restaurants to identify restaurant critics).

4.4. Secrecy/Confidentiality

It should be possible to make it infeasible for any outside party without knowledge of the shared secret being used to learn what information is exchanged using OWAMP-Control by inspecting an OWAMP-Control stream or actively modifying it.

(It is recognized that some information will inevitably leak from the mere fact of communication and from the presence and timing of concurrent and subsequent OWAMP-Test traffic.)

4.5. Integrity

So that it is possible to detect any interference during a conversation (other than the detention of some messages), facility must be provided to authenticate each message of the OWAMP-Control protocol, its attribution to a given session, and its exact placement in the sequence of control protocol exchanges.

It must also be possible to authenticate each message of the test protocol and its attribution to a specific session, so that modifications of OWAMP-Test messages can be detected. It must be possible to do this in a fashion that does not require timestamps themselves to be encrypted; in this case, security properties are valid only when an attacker cannot observe valid traffic between the OWAMP-Test sender and receiver.

4.6. Replay Attacks

OWAMP-Control must be resistant to any replay attacks.

OWAMP-Test, on the other hand, is a protocol for network measurement. One of the attributes of networks is packet duplication. OWAMP-Test has to be suitable for measurement of duplication. This would make it vulnerable to attacks that involve replaying a recent packet. For the recipient of such a packet it is impossible to determine whether the duplication is malicious or naturally occurring.

OWAMP-Test should measure all duplication -- malicious or otherwise. Note that this is similar to delay attacks: an attacker can hold up a packet for some short period of time and then release it to continue on its way to the recipient. There's no way such delay can be reliably distinguished from naturally occurring delay by the recipient.

OWAMP-Test should measure the network as it was. Note, however, that this does not prevent the data from being sanitized at a later stage of processing, analysis, or consumption. Some sanity checks (those that are deemed reliable and erring on the side of inclusion) should be performed by OWAMP-Test recipient immediately.

4.7. Modes of Operation

Since the protocol(s) will be used in widely varying circumstances using widely varying equipment, it is necessary to be able to support varying degrees of security modes of operation. The parameters to be considered include: confidentiality, data origin authentication, integrity and replay protection.

It should also be possible to operate in a mode where all security mechanisms are enabled and security objectives are realized to the fullest extent possible. We call this "encrypted mode".

Since timestamp encryption takes a certain amount of time, which may be hard to predict on some devices (with a time-sharing OS), a mode should be provided that is similar to encrypted mode, but in which timestamps are not encrypted. In this mode, all security properties of the encrypted mode that can be retained without timestamp encryption should be present. We call this "authenticated mode".

It should be possible to operate in a completely "open" mode, where no cryptographic security mechanisms are used. We call this "unauthenticated mode". In this mode, mandatory-to-use mechanisms must be specified that prevent the use of the protocol for network capacity starvation denial-of-service attacks (e.g., only sending test data back to the client that requested them to be sent with the request delivered over a TCP connection), and that prevent a worm from using the protocol to send test data to a very large number of hosts in a short time (e.g., ensuring that open mode requests can only be made by humans, rate-limiting the acceptance of open mode requests).

To make implementation more manageable, the number of other options and modes should be kept to the absolute practical minimum. Where choosing a single mechanism for achieving anything related to security is possible, such choice should be made at specification phase and be put into the standard.

5. IANA Considerations

Relevant IANA considerations will be placed into the protocol specification document itself, and not into the requirements document.

6. References

6.1. Normative References

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

[RFC2474] Nichols, K., Blake, S., Baker, F. and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.

[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.

[RFC3140] Black, D., Brim, S., Carpenter, B. and F. Le Faucheur, "Per Hop Behavior Identification Codes", RFC 3140, June 2001.

6.2. Informative References

[BRIX] Brix 1000 Verifier, http://www.brixnet.com/products/brix1000.html

[CQOS] CQOS Home Page, http://www.cqos.com/

[RIPE] RIPE NCC Test-Traffic Measurements home, http://www.ripe.net/test-traffic/

[SURVEYOR] Surveyor Home Page, http://www.advanced.org/surveyor/

7. Authors' Addresses

Stanislav Shalunov

EMail: shalunov@internet2.edu

Benjamin Teitelbaum

EMail: ben@internet2.edu

8. Full Copyright Statement

Intellectual Property

Acknowledgement

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