

LMAP
Internet-Draft
Intended status: Informational
Expires: March 25, 2013

H. Schulzrinne
W. Johnston
J. Miller
FCC
September 21, 2012

**Large-Scale Measurement of Broadband Performance: Use Cases,
Architecture and Protocol Requirements
draft-schulzrinne-lmap-requirements-00**

Abstract

Measuring broadband performance on a large scale is important for network diagnostics by providers and users, as well for as public policy. To conduct such measurements, user networks gather data, either on their own initiative or instructed by a measurement controller, and then upload the measurement results to a designated measurement server. This document describes a logical architecture and summarizes key requirements for protocols to connect the components. The system is designed to support residential and small-enterprise networks, using either wired or wireless networks. The architecture supports an extensible set of active and passive measurements, but the details of the metrics themselves are beyond the scope of this document.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

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

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

This Internet-Draft will expire on March 25, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	3
2.	Use Cases	6
3.	Architecture Overview	8
3.1.	Measurement client	8
3.2.	Measurement server	8
3.3.	Measurement controller	8
3.4.	Data collector	8
3.5.	Network parameter server	9
4.	Protocols	10
5.	Initiation of Measurements	12
6.	Requirements	13
7.	Security Considerations	16
8.	IANA Considerations	17
9.	Acknowledgements	18
10.	References	19
10.1.	Normative References	19
10.2.	Informative References	19
	Authors' Addresses	20

1. Introduction

Measuring actual network performance is crucial to managing consumer and enterprise networks, but, when performed at scale, it also allows third parties to gain insight into the actual performance of such networks, facilitating consumer choice and allowing to evaluate the state of broadband performance in a country, among other public policy goals. A number of network performance metrics have been defined, such as [2], but there is no overall architecture and set of protocols that facilitates gathering such measurements in a coordinated way, at scales drawing on thousands or millions of nodes.

Large-scale measurement efforts (e.g., [3]) use proprietary, custom-designed mechanisms to coordinate the measurement clients. They require that the organization running the measurements deploy thousands of dedicated hardware components or rely on end-system software modules that are subject to exogeneous factors, such as home networks, that may distort the results. Thus, this document proposes an overall architecture, with emphasis on the functional and security requirements for the protocols connecting the elements of the architecture, that will make it possible to build measurement capabilities into home and enterprise edge routers, personal computers, mobile devices and other edge devices.

Any usage and implementation will likely impose a number of additional operational requirements and a statistical sampling methodology. For example, the Measurement Broadband America project [3] within the US Federal Communications Commission (FCC) has established specific operational guidelines on data validity and commits to specific requirements for open access to measurement data, software tools and documentation of measurement methodology and statistical approaches. While crucial for deployment, these are beyond the scope of this protocol requirements document. Also, as is customary for IETF-managed protocols, this document does not mandate a specific hardware or operating system platform for implementation.

We suggest that the IETF IP Performance Metrics (IPPM) working group take on defining any additional performance metrics as needed. Such an effort should be undertaken as a collaborative effort with the Broadband Forum (BBF) [4]; other SDOs may also take on aspects of this problem area.

In some applications, such as data gathering by local regulatory entities, extensive logging at various levels, from packet arrival times to events, will be used to assure all parties of the validity of the data gathered. However, logging is beyond the scope of this document.

Both active and passive measurement techniques have been widely accepted in practice. In active measurements, the end systems emits traffic and observes a performance metric, or has another end point do so. Examples of active measurements include round-trip delay [2], one-way delay [5] and throughput [6] metrics, service availability, as well as a range of measurements that try to emulate application behavior, such as VoIP, HTTP retrievals or media streaming. Passive measurements observe existing user traffic flows. We note that there is some overlap between NetFlow [7] measurements and passive measurements described here. The delineation between the two and possible re-use of functionality are left to further discussion.

For both active and passive measurements, a measurement client sends or observes traffic, respectively. For active measurements, the measurement client may need a measurement server as serve as recipient of the measurement traffic. (In some cases, such as measurements modeling user access to network services, such as web page retrieval performance, the measurement traffic is exchanged with a production server, such as a web server, but this requires careful design to avoid overloading that server with measurement traffic.) Since we are interested in large-scale measurements, we assume that a measurement controller provides the measurement client with information on what to measure and when to perform the measurements. Finally, in some cases, a measurement data collector gathers data, typically samples rather than aggregate data, collected by the measurement clients for later analysis. The data models and file formats for supporting the exchange of the test parameters as well as test results require standardization.

As noted above, it appears likely that metrics will evolve and new ones will be added over time. Components of the platform may be designed and operated by different, independent entities, or, at minimum, data gathered by the platform may be used by different parties for different purposes. For example, a regulator or ISP might contract with third parties to manage various components of a measurement effort, and all data communications must securely support the delegation and authentication of rights and responsibilities to perform any operational parameter supported by the measurement architecture. Thus, it will be important to agree to on a set of metrics and associated metric-specific protocol parameters. For example, the TCP throughput metric defined in [6] depends on the TCP congestion avoidance algorithm. Each measurement run generates one or more data samples, e.g., a set of throughput values. The controller needs to convey those parameters to the measurement client and the data collector needs to be able to determine unambiguously which parameters were used for a specific set of data samples.

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 [[1](#)]. Although [RFC 2119](#) was written with protocols in mind, the key words are used in this document to indicate the strength of a requirement.

2. Use Cases

Large-scale, automated measurements are helpful in a number of use cases. We illustrate the scope with three examples:

Provider network measurements: Internet service providers have an interest in knowing how well their networks are performing, as viewed from their customers' perspective. Such performance information allows them to identify bottlenecks and observe the impact of changes in user behavior, e.g., the emergence of new network applications or time-of-day patterns. Here, the provider is not interested in the performance of an individual edge network or device, but rather wants to get a statistically-valid sample of performance across their network. Service providers may be interested in both the end device performance, i.e., the performance as seen by edge devices in home and enterprise networks, as well as the edge performance, i.e., as seen by the network device directly attached to their network, such as a cable modem, DSL modem or enterprise edge router. To reduce the network load, providers are unlikely to gather measurements from all clients all the time, but rather sample randomly across both time and their user population. The measurement controller directs the measurement client what measurements are to be performed, what measurement servers to use, when to measure and at which data collector it should deposit the measurement data.

User network diagnostics: End users may want to determine whether their network is performing according to the specifications (e.g., service level agreements) offered by the Internet service provider, or they may want to diagnose whether components of their network path are impaired. End users may perform measurements on their own, using the measurement infrastructure they provide or infrastructure offered by a third party, or they may work directly with their network or application provider to diagnose a specific performance problem. Depending on the circumstances, measurements may occur at specific pre-defined intervals, or may be triggered manually. A system administrator may perform such measurements on behalf of the user.

Multi-provider network measurements: As an extension of the first use case, multiple network providers and third parties, such as a regulatory body, may collaborate to gather network performance data on a one-time or recurring basis, using a subset of customers of the service providers. The form of collaboration is beyond the scope of this paper, however it should be understood that a data collection platform must serve multiple stakeholder interests.

In the description above, the network provider can either be a

commercial or not-for-profit entity distinct from the network edge users, or it can be the information technology department in a local area network. Particularly for the user diagnostics use case, it may be helpful for the measurement client to obtain parameters of their connectivity, such as the nominal uplink and downlink speed. In other cases, only the entity performing the data analysis may need to know the nominal performance parameters.

3. Architecture Overview

We define a measurement platform to consist of one or more measurement clients, measurement controllers and data collection servers. Based on the use cases above, we summarize their functions below.

3.1. Measurement client

The measurement client is the reference point for measurements. For active measurements, it sends measurement traffic to the measurement server or other network elements. For passive measurements, it observes network performance metrics. Client measurement functionality must be implementable in a variety of user contexts and provide for communications within different network segments, such as the access link between a broadband subscribers modem and an ISP network, as well as consumer electronic device communicating to measurement server features in a wireless LAN device.

3.2. Measurement server

The measurement server is only needed for active measurements that require two network nodes. The measurement server typically operates as a traffic source or sink. To allow scaling, different clients within a measurement platform may use different measurement servers. Clients may also select, for example, the closest measurement server if the influence of wide-area connectivity on measurement results is to be minimized.

3.3. Measurement controller

The measurement controller provides the measurement client with instructions on when and how to conduct what measurements, i.e., the measurement schedule. For example, it might instruct the client to conduct a particular kind of throughput measurement every ten minutes, and to deposit the throughput samples into a particular data collector. Measurement controllers may be capable of accepting inputs from other controllers, scaling up the scope of the measurement system. As one example, an ISP operating a testing platform for its own network may accept test requests from an external controller as part of a nationwide testing program that it is participating in.

3.4. Data collector

The data collector collects time-stamped measurement samples from measurement clients. It generally makes these measurement samples available only to authorized users. The data collector may store

measurement samples in a database or as files and may make them available via download or SQL query. Access control, internal data storage and access methods to data are beyond the scope of this document.

We logically separate the data collector from the measurement server for both functional and performance reasons. In general, data collected should not be transferred to the collector while a measurement is in progress. Also, a measurement client on a mobile host may decide to delay transferring measurement data until a low-cost or high-speed connection to the server becomes available.

3.5. Network parameter server

In some of the use cases, it is necessary for the analysis to compare the measured against the nominal network performance, or correlate measured parameters with the type and key parameters of the user's network connection. For example, for evaluating network delay measurements, it is helpful to know what kind of access technology (e.g., FTTP, DSL, cable, cellular data or satellite) and nominal speed the network connection offers.

4. Protocols

With the description of the elements above and the relationships between them, a set of protocols needs to be defined. The key functions of the protocols are described briefly below.

Measurement client to measurement server: Each metric will have its own set of measurement protocols, and these are beyond the scope of this document. For example, a VoIP metric may use a defined set of UDP packets to estimate performance.

Measurement client to measurement controller: The measurement client queries the measurement controller to obtain an updated measurement schedule. The measurement schedule returned by the controller indicates the type of measurements the measurement client should perform, the measurement servers and on what schedule to conduct the measurements. For example, it might indicate to run a VoIP emulation test every day for ten minutes to a specific server, spanning a one-week measurement campaign. The collector also indicates one or more addresses of data collectors to the client.

Measurement controller to measurement controller: A measurement controller can request that another controller undertake a specific testing program and could indicate specific tests, schedules and sample parameters appropriate to the intended objectives. Other data could include the identity and identity verification of the requester, a specific test identifier, e.g. Nationwide Test XX, and information necessary for the data collector so that data is accessible to authorized parties.

Measurement client to data collector: The measurement client will typically perform one or more measurements, and then, during the pause between measurements, transmit the collected samples to the data collector. The samples must be tagged with identifying information, such as when they were collected, edge device information (e.g., the mobile device or cable modem) and which measurement host was used. For mobile measurements, the sample data is likely to contain location data, possibly of reduced spatial resolution to protect user privacy.

Measurement client to network parameter server: The measurement client may query the network parameter server, typically located in the service providers network, for information about its nominal service parameters, based on its network address, link layer address, or hardware identifiers such as the IMEI for mobile nodes. The data returned may include information such as nominal uplink and downlink speeds, data quotas and physical and data link

layer technology. (Data quotas may be important for deciding which data-intensive measurements a client wishes to run.)

While basic network connection information is unlikely to change rapidly, it may change at unpredictable instants. For example, a network provider may upgrade the connection speed of subsets of their customers, customers may change their subscription or provider may adjust the monthly data transfer quota.

We assume that the measurement server, controller and data collector cooperate in configuring appropriate parameters. For example, the controller needs to be able to determine which measurement servers and data collectors are currently available and the client is authorized to use. Discovery of suitable data collectors is considered beyond the scope of this effort.

5. Initiation of Measurements

Either the client or the measurement controller could in principle initiate measurements. For periodic measurements or one-off user-triggered diagnostics, it is sufficient for the end system to contact the controller, e.g., periodically every week. Client-initiated measurements have a number of advantages. In particular, they make it less likely that measurement hosts can be abused to generate denial-of-service traffic. They also avoid problems allowing inbound requests through network address translators (NATs) and firewalls.

However, there may be cases where the network provider wishes to initiate a one-time measurement or change the measurement parameters before the client next contacts the controller. For such cases, a publish-subscribe mechanism may be considered, where the measurement client subscribes to measurement schedule updates with the measurement controller.

6. Requirements

We distinguish requirements for the different component by a prefix: Requirements labeled A-* describe the overall platform architecture, M-* indicate requirements primarily affecting the measurement client, C-* those for the controller, D-* for the data collector and N-* for the functions necessary to obtain network parameter. In many cases, a single requirement governs more than one entity or protocol, so the labeling should be considered rough.

A-1: The architecture MUST allow for one-time measurements initiated by end users, sampled measurements initiated by network providers and measurements by one or more third parties.

A-2: Measurement clients and servers MUST support an extensible set of performance metrics.

A-3: Measurement clients, measurement servers and data collectors MAY be operated by different administrative entities, including entities other than the Internet service provider.

A-4: Measurement clients MUST be able perform both active and passive measurements.

A-6: All entities MUST be able to authenticate the entities they communicate with.

A-7: Each measurement sample MUST be unambiguously associated with the measurement parameters, either by reference or by value.

A-8: To ensure availability and scaling, implementations MUST be able to implement multiple measurement controllers, measurement servers and data collectors with appropriate load balancing and failover.

M-1: The architecture MUST allow a single measurement client to participate in one or more independent measurement platforms.

M-2: A measurement client SHOULD be able to automatically switch from a non-responsive to an alternate measurement server.

M-3: A measurement client MUST be able to register with the data collection platform automatically, announcing its availability and relevant system parameters. (For example, a cable or DSL modem may indicate its make and model number.)

- M-4: A measurement client MUST be able to declare what kind of measurements it can perform, e.g., by enumerating a set of measurement identifiers.
- C-1: The measurement system MUST support measurements that are scheduled according to a pre-defined calendar.
- C-2: The measurement controller MUST be able to specify the interval on how often it wishes to be contacted for updated measurement schedules.
- C-3: A measurement client SHOULD be able to automatically discover controllers provided by their Internet service provider.
- C-4: A measurement client MUST be able to authenticate and authorize the measurement controller.
- C-5: The data exchange between the client and controller MUST allow for optional encryption and integrity protection.
- D-1: The protocol messages for measurement samples MUST allow new measurement types and parameters.
- D-2: It MUST be possible to protect the integrity and confidentiality of the measurement data exchanged between the measurement client and the data collector.
- D-3: The data exchange protocol between measurement server and data collector SHOULD allow the definition of common data elements, e.g., for network addresses and timestamps.
- D-4: The measurement client SHOULD be able to automatically fail over to alternate data collectors.
- D-5: Clients MUST be able to either send data immediate or delay sending measurement data to the collector, e.g., to use a low-traffic period or a low-cost network.
- D-6: Clients MUST be able to interleave data samples from different measurement metrics to the data collector.
- D-7: The data collector SHOULD be able to ascertain whether the measurement client clock is at least approximately synchronized to its own.

D-8: The data exchange between measurement client and data collector MUST be subject to flow and congestion control.

D-9: The measurement client MUST be able to ascertain that it is initiating a session with the desired data collector rather than an impostor.

N-1: Measurement clients SHOULD be able to obtain nominal network service parameters in a machine-readable format, such as advertised speed and typical latency. (This may not be necessary in all measurement use cases.)

N-2: The set of network parameters MUST be extensible in a backward-compatible manner.

N-3: The measurement client SHOULD be able to determine the network parameter server without manual configuration.

N-4: The protocol between measurement client and network parameter server SHOULD support a variety of client identifiers, such as network addresses, link-layer addresses, AAA identifiers or hardware identifiers.

N-5: The data exchanged between the network parameter server and the measurement client SHOULD ensure its confidentiality and integrity.

N-6: The protocol SHOULD support suitable authentication functionality to restrict access to network parameters to authorized nodes. Authorized nodes may include third parties, such as data collectors.

N-7: The entity querying the network parameter server MUST be able to assure itself that it is communicating with an authentic server.

N-8: Clients of the network parameter server SHOULD be able to be automatically informed of changes in parameters.

7. Security Considerations

The large-scale measurement architecture has to prevent third parties' use of the measurement clients in bot-nets or for other nefarious or malicious purposes. A malicious third party could cause a measurement client to initiate probe traffic to victim hosts rather than measurement servers. We rely on user-initiated requests, secured with transport-layer security and server certificates, to ensure that only user-authorized entities issue control commands. Users may also authenticate themselves via local shared secrets. We note that there are similarities in approach with M2M data communications and we suggest that reference of ongoing work on the M2M signaling gateway framework or other models may be useful.

Measurements may also inadvertently expose information that the owner of the measurement client considers privacy-sensitive. Privacy considerations may differ depending on whether the measurement client, measurement server or data collector are operated by the same entity or not, and what trust relationships these entities have with each other. It must be possible to protect the confidentiality of the measurement data exchanged between the measurement client and the data collector. For mobile measurements, location information is likely to be crucial to interpreting measurement results. A measurement client may want to substitute rough location [8] to reduce the ability of a third party to track its movements and whereabouts.

8. IANA Considerations

This document does not request any IANA actions.

9. Acknowledgements

The document is based on discussion within the FCC Measuring Broadband America project.

DISCLAIMER: The opinions expressed are those of the author and do not necessarily represent the views of the Federal Communications Commission or the United States Government

10. References

10.1. Normative References

- [1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [RFC 2119](#), March 1997.

10.2. Informative References

- [2] Almes, G., Kalidindi, S., and M. Zekauskas, "A Round-trip Delay Metric for IPPM", [RFC 2681](#), September 1999.
- [3] Federal Communications Commission, "Measuring Broadband America", September 2012.
- [4] Broadband Forum, "Liaison Statement: New Project - Broadband Access Service Attributes and Performance Measures", August 2012.
- [5] Almes, G., Kalidindi, S., and M. Zekauskas, "A One-way Delay Metric for IPPM", [RFC 2679](#), September 1999.
- [6] Mathis, M. and M. Allman, "A Framework for Defining Empirical Bulk Transfer Capacity Metrics", [RFC 3148](#), July 2001.
- [7] Claise, B., "Cisco Systems NetFlow Services Export Version 9", [RFC 3954](#), October 2004.
- [8] Barnes, R. and M. Lepinski, "Using Imprecise Location for Emergency Context Resolution", Internet draft [draft-ietf-ecrit-rough-loc-05](#), July 2012.

Authors' Addresses

Henning Schulzrinne
Federal Communications Commission - See Disclaimer
445 12th Street SW
Washington, DC 20554
USA

Phone: +1 202 418 1544
Email: henning.schulzrinne@fcc.gov

Walter Johnston
Federal Communications - See Disclaimer
445 12th Street SW
Washington, DC 20554
USA

Phone: +1 202 418 0807
Email: walter.johnston@fcc.gov

James Miller
Federal Communications Commission - See Disclaimer
445 12th Street SW
Washington, DC 20554
USA

Phone: +1 202 418 7351
Email: James.Miller@fcc.gov

