IPPM Working Group
Internet-Draft
Intended status: Standards Track
Expires: 11 September 2026

T. J. Pinkert
N. Masoudifar
A. A. Nnamdi
Siemens Mobility GmbH

10 March 2026

An IPv4 and IPv6 measurement option (MO) for passive flow measurements. draft-pinkert-ippm-ip-measurement-option-00

Abstract

This document introduces an internet protocol (IP) measurement option (MO) that contains information, normally not available to the receiver of IP packets, to perform passive network measurements. In particular, measurements that need a sender time stamp and a packet order number are then possible directly at the receiver. The information contained in the IP MO can also be used by hosts en-route to perform slightly more limited passive network measurements.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://example.com/LATEST. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-pinkert-ippm-ip-measurement-option/.

Discussion of this document takes place on the ippm Working Group mailing list (mailto:WG@example.com), which is archived at https://example.com/WG.

Source for this draft and an issue tracker can be found at https://github.com/USER/REPO.

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 https://datatracker.ietf.org/drafts/current/.

Copyright Notice

Table of Contents

1. Introduction

Passive network measurements on differentiated services (DS) microflows [RFC2475], following the IP performance metrics (IPPM) rationale outlined in [RFC2330], can be performed for a limited number of metrics without the need for P-type packets. Using passive measurements, a complete characterisation of an end-to-end path through the internet is not possible. Nevertheless, passive measurements allow a measurement host to obtain important information about the minimum or maximum quality (depending on the situation and metric) that the network currently offers.

The P-type packets used by, e.g., the one-way active measurement protocol (OWAMP, [RFC4656]) and two-way active measurement protocol (TWAMP, [RFC5357]), contain additional information that is needed for the calculation of certain metrics. Two important pieces of information are the sequence number and the timestamp, that are needed to calculate packet loss metrics and packet delay metrics.

To include such information in an IP packet, for measurement at the IP layer, a suitable solution needs to be found. Since the data of IP packets is determined by the upper stack configuration and the application protocol, the only way is to include this information in the IP header. Fortunately, the designers of the IPv4 and IPv6 protocols acknowledged the possibility of inclusion of information for various purposes in the IP header as IP options.

This document describes an IPv4 and IPv6 measurement option (MO) for the measurement of packet loss, packet delay and other metrics that require a sequence number and a timestamp from the sending host at the receiving host.

2. Justification for an IP measurement option

In internetworks, the IP protocol is the unifying protocol in the network stack. At the link layer various network technologies can be deployed that are all capable of transporting IP packets. Although measurement techniques and specifications are present for the lower layers, it is typically hard to implement these such, that data from specific applications on a host can be measured.

Another concern is that, at the data link layer, each particular link layer protocol adds protocol control information with a technology dependent amount of data, to each IP packet. Therefore, measurements performed on various data link layers cannot always be compared without assumptions.

Typical internet applications use "sockets", a concept found in many of the major operating systems, where both IP addresses, the transport protocol and its port numbers are grouped. This is exactly the definition of a microflow.

Nevertheless, transport layer protocols do typically not have the flexibility to allow for measurements at the transport layer, since not all, or maybe even least of the, transport protocols, include a sequence number and timestamp in their data fragments. Also at the transport layer, the argument holds that measurements are not necessarily comparable between different transport protocols. In addition, a generic software, would also need adaptation for each transport protocol, to perform (comparable) measurements when operating higher up in the network stack.

The IP protocol gets part of its flexibility from the capability of inserting IP options into the IP header. Together with its unifying properties on internet hosts, which allows comparison of measurement data among applications and hosts, the network layer with the IP protocol forms the ideal place for measurements, as acknowledged by the IPPM framework. The only downside of an IP measurement option is that the IP header is enlarged. Adding slightly to the amount of data sent through the network. In many cases this is offset by the benefits of being capable of assessing the network quality at real-time.

3. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

4. Metrics for the IP measurement option

The IP measurement option will be designed specifically for the following metrics: - One-way Packet Delay, Mean Packet Delay, Packet Delay Variation, - One-way Packet Loss, Packet Reordering and Packet Duplication.

The measurement option must be suitable for the measurement of microflows as defined by [RFC2475] in the scope of DS. Microflows are identified by IP source and destination addresses, the protocol id and transport layer source and destination port numbers (when available). Measurements must be possible between two end hosts on a point-to-point (P2P) IP link. Extension to multi- point-to-multi-point links (MP2MP) should be possible. This requires that each packet can be uniquely assigned to a microflow.

In addition, it should be possible that hosts en-route, can use the IP MO to measure or estimate the metrics on a microflow basis. Such measurements can be done at boundary / gateway hosts of the network. One difficulty, in this case, can be, that not all packets of a microflow must necessarily pass a certain host, since paths through the network, for packets of the same micro- flow, may differ.

5. Considerations regarding the option content

Data rates and packet rates per microflow can be readily measured by source, observer, and destination using a clock and the number of octets per microflow traversing the network interface. These metrics do not need additional information.

5.1. Concerning loss, reordering and duplicates

To measure packet losses, reordering and duplication, each packet must be uniquely identifiable, and orderable, within the microflow. Only selected transport layer protocols include such information, e.g., the TCP protocol.

* The IP MO must include information to uniquely identify the position of each IP packet in a stream of packets.

5.2. Concerning network traversal time (delay)

To measure packet delay and other time-based properties of the network, the time at the start of transmission (wire arrival time at the transmitting host) and end of reception (wire exit time at the receiving host) of the IP packet must be recorded. It is assumed in this document that hosts hosts have well synchronised clocks. Clock synchronization is a topic that is addressed abundantly elsewhere

([RFC2330]).

The sending host records the start of transmission time, while the receiving host records the end of reception time. In combination with the network interface speed and the number of octets in the IP packet, the start of reception time can be calculated at the receiving host. Note that the receiving host can only timestamp at the end of reception, because only then is clear that the packet has arrived complete and in good order.

An exemption here can be hardware timestamp units that mark the start of reception time. However, this time may include the store and forward time of the input queue. With the wire arrival time at both hosts, the network delay can be determined.

The timestamps for start of transmission (transmission time stamp, TTS) and start of reception (reception time stamp, RTS), as recorded in the host software, may need correction to accurately represent the wire arrival times. Such corrections are common in time transfer systems. The correction problem however does not change the requirement that the wire arrival time of the source needs to be sent to the receiver.

* The IP MO must contain the wire arrival time of the sender.

5.3. Concerning microflow identification

Although it is assumed that microflows form a single data stream, this must not be the case. Consider a user of a computer starting and stopping a group of network programs at the same port, contacting the same destination host at the same port with the same protocol. It is now unclear, since multiple programs were used, if the measured data should be grouped in one measurement or not. Another use case is that where the higher layer protocols do not contain port numbers in the "classical" sense (classical transport protocols are UDP, TCP and other internet transport protocols).

This ambiguity is resolved by the inclusion of a flow label in the IPv6 protocol. A unique flow identifier is assigned, by the sending host, to packets belonging to the same microflow.

* The IPv4 MO must contain a flow identifier in equivalence to the IPv6 protocol.

5.4. Concerning packet selection for measurement

To apply "alternate marker" methods [RFC9341], there must be a way to apply a (single bit) marker in the IP header. This can be done using the DSCP [RFC2474] field of the IP packet, and alternating two code points while keeping the transport properties for both code points equal, or by abusing the single bit that is still free in the CU fields specified in [RFC2474]. The first would need broader support to implement the method, the latter would void the internet standards. Notably the ECN [RFC3168] and L4S specifications [RFC9330], [RFC9331], [RFC9332] that uses these bits.

* The MO must contain a marker to enable the use of alternate marker methods.

Another consideration is that measurement software may not be capable of processing all packets on high-speed interfaces. In this case, either layer 3 hardware support would be needed, or statistics could be gathered on only a fraction of the packets that are sent. In the latter case it would be unfavourable to apply an MO only to the packets that should be measured, because network components may react differently on packets of different lengths, and, potentially, on packets with different header lengths.

* The MO must contain a marker to signal which packets shall be used to calculate metrics on.

5.5. Concerning integrity of measurement data

Since the network traffic with IP measurement options included may pass network nodes that are owned by foreign entities, the IP measurement option must be protected against manipulation. One way to do this is to include a digital signature into the MO option. Another way to do this is to encrypt the contents of the measurement option for all hosts that should not read its contents.

* The MO must contain the possibility to include a cryptographic signature.

* Encryption of the MO contents, potentially including a cryptographic signature, must be possible.

5.6. Concerning fragmentation of IP packets

Fragmentation of IP packets works as follows. The IP header of a packet that needs fragmentation, is copied into each fragment. The data of the IP packet is cut into slices, and each slice is added to a new fragment. Then the more fragments flag and the fragment offset fields are set / updated. The more fragments flag is set to zero on the last fragment.

The identifier field is used to distinguish fragments of more than one package. The sending host must ensure that the same identifier value is not reused when packets can still be under way in the network (network timeout, 255 seconds). Due to this reason, modern layer 4 protocols, notoriously the TCP protocol try to avoid that IP packets are fragmented, since only 64k fragments can be underway in an IP network at the same time. It can be understood that with the current higher speed network interfaces, this poses a limitation on the data throughput possible for a single microflow.

Nevertheless it is good practice to consider fragmentation in the design of an IP measurement option. The specification of IPv4 options provides the copy flag, that allow an option to be present in the first fragment only, or to copy it into all fragments.

Since our IP measurement option could still be used to obtain information on the network performance by hosts enroute, when it is present on fragments of an IP datagram, it would be favourable to copy the data of the IP measurement option into each fragment.

* The IP MO must be included in all fragments of an IP datagram.

Fragments of a datagram can be identified by the identifier, fragmented flag, and the contents of the fragment offset field. This may help enroute hosts to interpret "the same" IP measurement option as not belonging to a packet duplicate. This also allows to measure fragment duplications both on the enroute host, as well as on the receiving host, that can be introduced as measurement additions. In this way fragment delay, loss, reordering or duplication statistics can be determined in addition.

The main measurements are to be performed on the source and destination hosts before fragmentation is performed and after re-assembly of fragments. This also means that measurements by systems "on the path" may yield different results than those on the end systems, unless they are capable of marking whether all fragments have passed such a system.

Since IPv4 packets, in contrast to IPv6 packets, can be fragmented by hosts enroute, fragmentation statistics may differ for various network segments.

6. Determination of the properties of the option content

In this section the contents of the MO determined above is specified specifically, based on the available technology and boundaries of the possibilities that an IP option offers. The following items were identified.

* The IP MO must contain the wire arrival time of the sender.
* The IP MO must include information to uniquely identify the position of each IP packet in a stream of packets.
* The IPv4 MO must contain a flow identifier in equivalence to the IPv6 protocol.
* The MO must contain a marker to enable the use of alternate marker methods.
* The MO must contain a marker to signal which packets shall be used to calculate metrics on.
* The MO must contain the possibility to include a cryptographic signature.
* Encryption of the MO contents, potentially including a cryptographic signature, must be possible.
* The IP MO must be included in all fragments of an IP datagram.

6.1. Current state of the art technology

The fastest data links today measure 1 Tbit/s per channel or more [Nokia]. It is assumed that 10 Tbit/s data link technology will become feasible in the near future. Links based on [IEEE802.3] Ethernet have a minimum frame size of 46 octets and a packet size of 72 octets. A 12 octets inter-frame gap must be added. The time to send the data on a 10 Tbit/s link can now be calculated. The number of bit times for one frame is 672. Each bit takes 0.1 picoseconds to send, therefore sending the minimum length frame takes 67.2 picoseconds.

On a 1 Pbit/s link, under the assumption that Ethernet frames are used, the time to send one minimum-length frame would be 0.672 picoseconds. At the latter rate, approx. 1488.1 Ethernet packets can be sent in 1 nanosecond.

At the network layer, the maximum datagram lifetime (MDL) of IP packets of is approximately 120 seconds [RFC6864].

The IPv4 protocol allows an IP header length of 15 * 32 bits. 10 * 32 bits are available for IP options. This is a total of 40 octets of which 38 can be used for user content.

The IPv6 protocol allows for 258 octet options of which 256 can be used for user contents.

Regarding time transfer techniques, two standards exist. The [IEEE1588] precision time protocol version 3 (PTPv3) is concipated to work with Ethernet, the IPv4 and IPv6 protocols, and the UDP protocol. The network time protocol version 4 (NTPv4) [RFC5905] is used in the scope of IPv4 and IPv6. The PTP protocol favours integer calculation, while the NTP protocol favours binary floating point calculations.

Time in the Linux OS kernel is delivered in nanoseconds resolution by the kernel interface. Time of the MAC OS X mach microkernel is delivered with nanosecond resolution as well. Time of the Windows kernel is delivered with a 100 nanosecond resolution.

6.2. Wire arrival time of the sender

The sender wire arrival time of the IP packet must be included in the IP option, such that the receiver can calculate the packet delay based on the receiver wire arrival time of the IP packet. Alternatively, both sender and receiver could send the wire arrival time to a third party that can then calculate the packet delay.

For the definition of time the major question is which resolution is sufficient. The next question is that of the calculator system. Is calculation done in integer of floating-point arithmetic. Integer calculation can be done on most machines, floating point arithmetic is expensive, both in hardware as well as in software when emulated.

Both in the NTP as well as the PTP protocol, the coarse timescale is kept in seconds from the epoch. The PTP protocol reserves 48 bits for the coarse timescale, the NTP protocol at least 32 bits, with the possibility of extending to 64 bits (the latter is more than enough for all practical purposes). The 16-bit format is typically only used for time differences.

It must be noted that, since the maximum datagram lifetime of an IP packet is 120 seconds, together with the fact that clocks in modern systems are synchronized with one second accuracy, the 7 least significant bits of the second counter of the 32- or 48-bit second counter, related to the epoch, should normally suffice to uniquely complete the timestamp based on the second counter of the receiver clock. Under the assumption that normal IP connections are lost within 255 seconds (TTL reduced to 0) an 8-bit seconds counter would suffice, it is mentioned in [RFC6864] that this is not always the case. To account for links where higher MDLs are possible an arbitrary number of bits could be added to this number. A 12-bit seconds counter would allow to identify delayed packets uniquely up to 1 hour with clock deviations on the systems of up to 150 seconds. This seems sufficient for most earth based purposes.

Only when the need to account for (inter)stellar applications arises, transfer of a larger part of the second counter would be needed. A 32-bit seconds timestamp allows for link delays of 136 years, which is sufficient for delay measurements for space travel within the solar system.

The fractional part of the second is encoded as a number of fractions in the NTP protocol. Each fraction is 2^(-N) long, where N is the number of bits used to encode the fractional seconds. NTP allows 16, 32 or 64 bits fractional seconds. The PTP protocol, on the other hand, encodes the fractional seconds part as a 32-bit number representing the number of nanoseconds in the fraction. Since most binary floating-point numbers have no exact match when converting to decimal floating-point numbers, rounding errors must occur when converting back and forth between NTP and PTP fractions. Reasons why to choose one format over the other must thus come from elsewhere.

A reason to favour the NTP format, may lie in the simplicity of certain calculations. Conversions between 16-, 32-, and 64-bit values can be performed by efficient bitshift operations. Adding and subtracting is also simply possible. A potential drawback are that rounding errors to nanoseconds exist, and those may not be distributed evenly. This may prevent the use of accurate high resolution measurements, where rounding errors may become significant.

One reason to favour the PTP format, which counts the fraction of time in an integer number of nanoseconds, lies in the field of frequency metrology. Many physical clocks count time based on a 10 MHz, 100 MHz, 1 GHz or even 10 GHz oscillator. These frequencies can be expressed as powers of 10 of a nanosecond. In addition, calculations in nanoseconds have a more natural feel than those in, e.g., 2^(-32) = 232.8... picosecond units. Other arguments are that

PTP implementations often feature accurate hardware timestamping units, and that all major operating systems deliver time in nanoseconds. Therefore the fractional seconds will be represented in nanoseconds.

Due to the technical restrictions of an IP option, the format for the timestamp must be chosen in a clever way. To represent one second in nanoseconds, 30 bits are sufficient. It is proposed to take the thirty least significant bits of a 32-bit word for this. Reserving the two most significant bits for other purposes. For the seconds part of the timestamp, 7 transferred bits are sufficient to reproduce the sender timestamp on the receiver, it would make sense to transfer at least an octet second part. The additional number of seconds may be used to relieve the requirement on the time synchronization of the two systems in use.

* For the IP measurement option, the PTP Timestamp format will be used as the basis for both seconds and nanoseconds. The least significant bits of the seconds field will be used in the IP measurement option.

6.3. Unique packet identifier

The classical means of determining whether packets are sent in order, is by use of a counter that increments upon sending of each subsequent packet.

As discussed in 5.2 each packet will be marked with a sender timestamp with ns resolution. On 10 Gbit/s links, sending one minimum size frame takes 67.2 ns. Therefore, the timestamp alone suffices to uniquely order packets sent on links up to 100 Gbit/s (6.72 ns / frame). At higher link speeds, an additional numerator can be used to subdivide the timestamp. For a 1 Pbit/s link, a numerator that is capable of counting to 1489 is sufficient to uniquely numerate ethernet frames and thus IP packets. This requires at least an 11 bit counter.

The IP ID field seems a good numerator candidate for counting datagrams, but it is currently specified to be solely used for fragmentation and reassembly by [RFC 6864]. Therefore, it is not suitable for measurement purposes (it is always zero for unfragmented packets) also because some sources do not vary the ID at all.

Duplication detection may be based on hashes [RFC 6621], but loss and reordering measurements are not possible when higher protocol layers lack information to detect these. This is e.g. the case with UDP. A timestamp can thus not replace a numerator field, since it does not show whether packets are missing, even on slow links.

A unique packet identifier of at least 12 bits is therefore recommended. Only on very fast links problems may occur when the counter overflows and multiple packet have the same timestamp and unique packet identifier.

6.4. Flow identifier

The last bit of contents that may be needed for transport layer-less protocols is a definition of a microflow identifier. For most current day protocols there are 16-bit port numbers specified for source and destination ports. This would call for a 32-bit source/ destination flow field. On the other hand, there are very few protocols without a transport layer. The IP packet definition allows only 256 of them, defined by the protocol byte. For outgoing flows, a host can only open 2^16 ports, so a 16-bit flow identifier for transport-less protocols suffices from this perspective.

The IPv6 protocol [RFC8200] specifies a 20-bit flow label. Defining 20 bits for the flow label allows unification of its use with IPv6 [RFC8200].

It must be noted that flow labels and the IP options are not protected against unwanted modifications and are for measurement purposes only. Transport-less protocols can open multiple data flows based on higher level protocols (e.g. application protocol). It is advised to generate flow labels stateful according to [RFC6437]. For compatibility reasons this will be the course of action for the IPv4 option.

6.5. Alternate marker

Passive measurements may be done using an alternate marker method such as specified in [RFC9341]. The measurement option could be a way to include such a marker independent from, e.g., an alternating DSCP. Therefore, at least one bit must be reserved as alternating marker bit.

6.6. Include in calculation marker

Since switches and routers may have differentiating behaviour when packet sizes differ, the same packet with and without the IP measurement option could be treated differently. To allow for unbiased measurements, even when only fractions of the packets of a microflow are to be included in measurements, the packets that are not to be measured on, must be of the same length. This means that a dummy measurement option must be included in the IP header in this case. At least one bit must be reserved as "include in calculation" bit, to allow the receiver to decide whether to include a packet in

the measurements or not.

6.7. Security measures

There are reasons to apply security measures to this data in the packet header. Since the IP measurement option can be used to verify an SLA / SLS, there may be monetary reasons to falsify the information in the packet, to make the measurement system think that all is fine on the network. E.g., forging of the timestamp by the last switch in the path may make the receiver think that the packet delay was less than it actually is.

There are multiple scenarios thinkable how IT security measures are applied to the IP measurement option.

1. Proof of integrity of the IP measurement option, while keeping the content available for third parties to use.

2. Encryption of the contents of the IP measurement option to hide the content for third parties.

Since the IP header, including the IP measurement option can be freely manipulated by hosts en-route, it is not possible to avoid that hosts may remove the IP measurement option entirely. All fields that need to be manipulated are easily adapted. An end-host may know that the IP measurement option has been removed, by being configured to expect the IP MO on certain connections with other hosts and not receiving any. The configuration can be static or dynamic, e.g., by means of a measurement management protocol. Such protocols would need definition in a separate RFC.

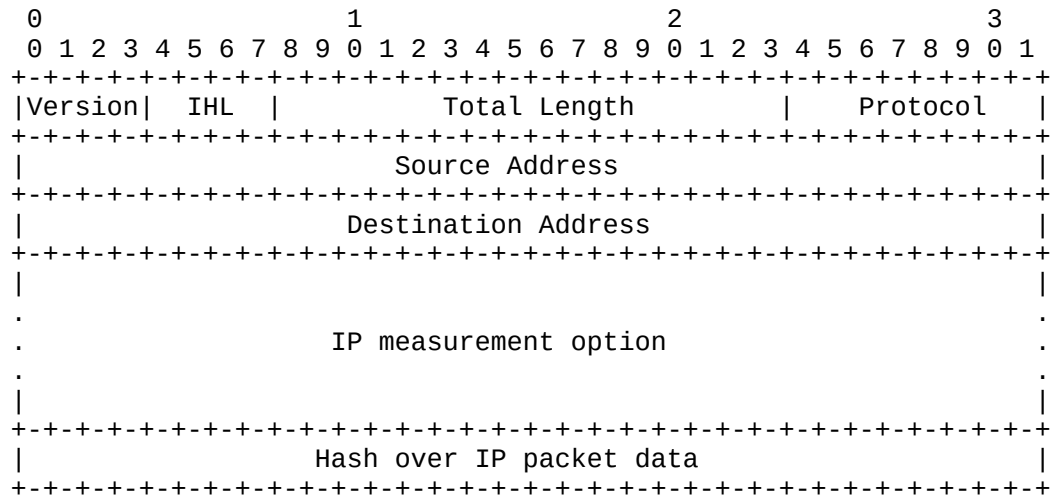
To ensure the integrity of the IP measurement option, a cryptographic signature can be used. It could be added as additional bytes to the IP measurement option. A HMAC or a hash of the signature can be added when symmetric keys are used. The latter has the advantage that the space needed to add the hash can be limited, e.g., by including only the first x octets of the full hash. It is quite expensive to add a full public/private key signature.

Upon signing an option, hosts that know the appropriate keys can verify the data in the IP measurement option. Depending on the type of signature used, it may be possible for hosts to change the option content and resign. It must, therefore, be ensured that, for such types of signatures, only trusted hosts obtain the keys. One indication of the presence of a signature is the length of the option. Hosts involved in verifying options before making measurements should however rely on other sources to determine if they expect signed measurement options. In this case they may also reliably detect removal of signatures, which is easily possible.

Fragmentation must be taken into account, and data in the IP header that is required for fragmentation and defragmentation may be changed by systems "on the path". It can therefore not be used as input for cryptographic signatures.

To perform various measurements the signature must be made over a pseudo header including the following data: 1. IP version 2. IP internet header length (Fixed value, 10 words) 3. IP total length (Payload length) 4. IP protocol type (Next header field) 4a. (Flow Label) 5. IP source address 6. IP destination address 7. IP measurement option, including: a. IP option type b. IP option length c. IP measurement option data excluding the signature / signature hash. 8. Hash over the IP packet data. (Hash over Payload)

The fields in the pseudo header are composed as follows for the IPv4 protocol:



The fields in the pseudo header are composed as follows for the IPv6 protocol:

The hash over the IP packet data (Payload) ensures that port numbers are always included when present, independent of the higher layer protocol, since it is favoured to make the implementation independent from possible higher layer protocols. The requirement on the hash over the IP packet data is that it can be quickly calculated, and that spoofing the data takes, on average, more time than the network timeout. Alternatively, a cryptographic hash, using a private key, may be derived over the pseudo header and the IP packet data (Payload). Note that the length of the hash must not be limited to 32 bits.

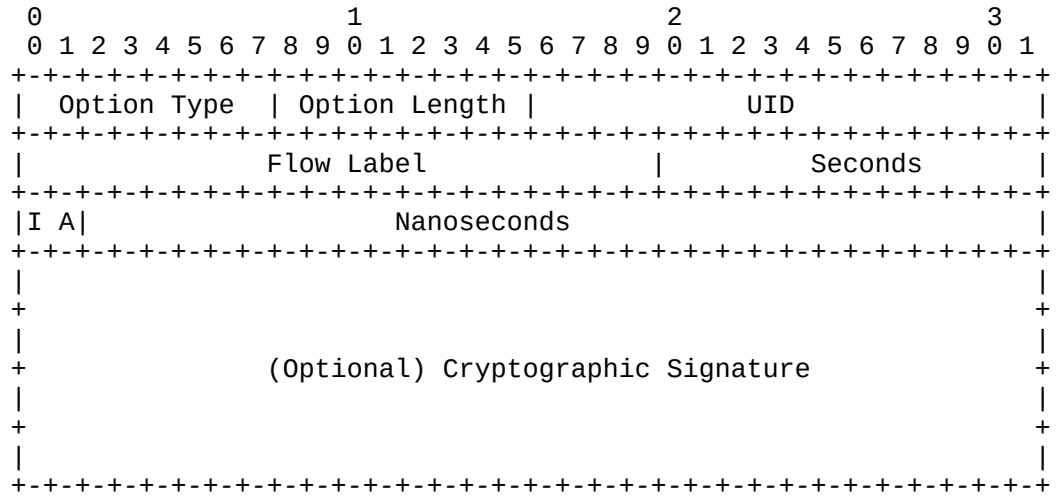
If cryptographic encryption is feasible given the maximum length of 40 octets (320 bits) of the IP MO, should be determined elsewhere. Also, which algorithms to use are not part of this specification. In any case it would be proper that untrusted hosts on the path know that options are encrypted, and cannot be used to perform measurements. Therefore, a way to determine whether, or not, an encrypted measurement option is used must be present. It is proposed to use the option type for these purposes.

In all cases the generation and distribution of cryptographic keys is performed by use of a secure key distribution protocol, that is not part of this specification. This protocol must be capable of generating keys for each distinguishable microflow to perform signing or encrypting the IP measurement option data.

The possibility of signed and encrypted IP measurement options is indicated in this RFC, but no particular specification is given yet. Such specifications are to be subject of RFCs to be written, as are the specifications for measurement management protocols.

7. IPv4 option definition

The maximum length of an IPv4 option is ten 32-bit words, or 40 octets, see [RFC791]. An IPv4 option always includes the option type. The length octet is always present for IPv4 options with more than only the option type. The IPv4 measurement option is therefore defined as follows.



Option Type: 8 bits The IP measurement option is a debugging and measurement option, and must be present in each fragment of an IP datagram. Two option types are requested, one for non-encrypted options, with or without signature, and one for encrypted options, that can only be read by systems on the path that belong to the trusted group of measurement clients.

Bit 0 (copied flag): 1 = copied Bit 1-2 (option class: 2 = debugging and measurement Bits 3-7 (option number): 26 = Unencrypted Measurement Option (UMO), to be assigned by IANA Bits 3-7 (option number): 27 = Encrypted Measurement Option (EMO), to be assigned by IANA

Option Length: 8 bits The length of the IP measurement option in octets, including length of the cryptographic signature when present. For an encrypted and signed measurement option, all octets are included in the length.

Unique Identifier (UID): 16 bits The Unique Identifier is incremented for each sent IP packet. This allows the receiving host to identify packet losses, duplications and re-orderings. It must be noted that each fragment of a fragmented datagram contains a copy of the IP measurement option. Therefore systems "on the path" should rely, e.g., on the fragment offset to perform such measurements. End-hosts perform measurements on the packet before fragmenting and after re-assembly.

Flow Label: 20 bits The Flow Label as specified in the IPv6 standard [RFC8200].

Seconds: 12 bits The 12 least significant bits of the seconds field of the PTP Timestamp.

Usage Flags: 2 bits The Usage Flags indicate how the option can be used by the receiving host.

Bit 0: 0 = Dont include in measurement’, 1 = Include in measurement
Bit 1: Alternate Marker

The I(nclude in measurement) flag signals whether or not the IP measurement option contains real data. Sending empty data (all 0) may be done to relieve computational efforts of the sending systems under high data traffic loads, while keeping the IP header size equal for all IP packets. For signed connections the cryptographic signature must also be applied on "empty" IP measurement options, since spoofing would otherwise be possible.

The A(lternate Marker) can be used to implement alternate marker measurement methods. Alternate marker methods using this IP measurement option, must specify how they use this bit.

Nanoseconds: 30 bits The nanoseconds field contains the number of nanoseconds of the timestamp. When a system is not capable of providing timestamps with nanosecond resolution, the highest resolution that can be provided is used and the other digits of the nanosecond field are set to 0. When a system uses, e.g., a 16-bit subdivision of one second (such as NTP timestamps may), the decimal value is rounded to the nearest nanosecond. Adding or subtracting a certain random number of nanoseconds to the result may be used to prevent rounding bias. This is up to the implementer of the IP measurement option for a system, but must be documented.

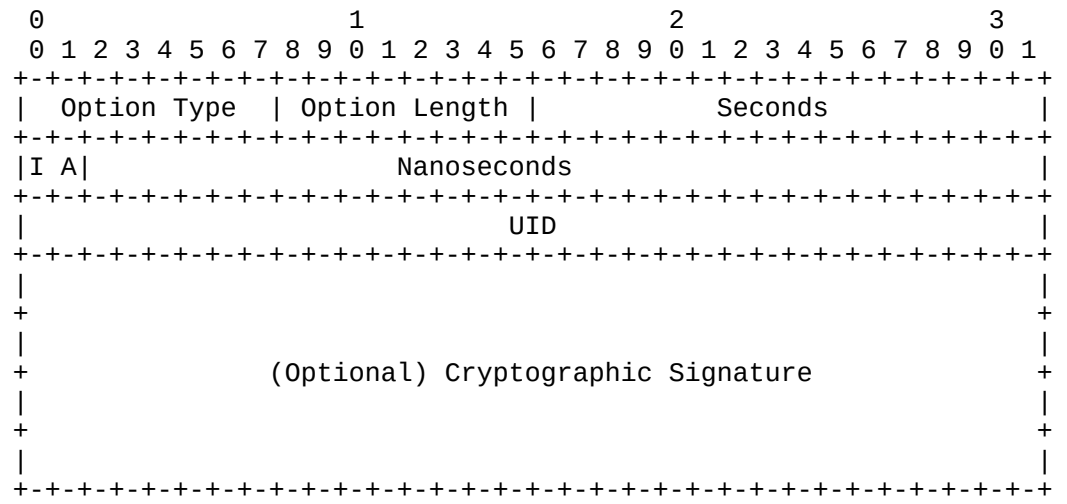
(Optional) Cryptographic Signature: Optionally up to 7 words, 28 octets, or 224 bits can be used for the transmission of a cryptographic signature.

The option order has been chosen such that when this option is the first option, the data aligns in the IP packet, hopefully resulting in an efficient processing in host systems, which would be needed on high-speed links. No EOOL option is needed when this option is the only one [RFC791], [IANA-IP-Option-Numbers].

When the option is encrypted only the first two octets have their normal meaning. The rest of the octets form the cyphertext.

8. IPv6 option definition

The maximum length of an IPv6 Hop-by-Hop Option is 255 octets, see [RFC8200]. An IPv6 option always includes the option type, and length octet. The IPv6 measurement option will be defined as follows.



Option Type: 8 bits The option may be skipped over if the Option Type is unknown, and the option data does not change en-route.

Bit 0-1: 00, Skip over this option and continue processing the header Bit 2: 0, Option Data does not change en route Bits 3-7: (option number): 26 = Unencrypted Measurement Option (UMO), to be assigned by IANA Bits 3-7: (option number): 27 = Encrypted Measurement Option (EMO), to be assigned by IANA

Option Length: 8 bits The length of the IP measurement option in octets, including length of the cryptographic signature when present. For an encrypted and signed measurement option all octets are included in the length.

Seconds: 16 bits The 16 least significant bits of the seconds field of the PTP Timestamp. When the IPv4 algorithms are used for calculations, the four most significant bits of the Seconds field may be skipped.

Usage Flags: 2 bits The Usage Flags indicate how the option can be used by the receiving host.

Bit 0: 0 = Dont include in measurement’, 1 = Include in measurement
Bit 1: Alternate Marker

The I(nclude in measurement) flag signals whether or not the IP measurement option contains real data. Sending empty data (all 0) may be done to relieve computational efforts of the sending systems under high data traffic loads, while keeping the IP header size equal for all IP packets. For signed connections the cryptographic signature must also be applied on "empty" IP measurement options.

The A(lternate Marker) can be used to implement alternate marker measurement methods. Alternate marker methods using this IP measurement option, must specify how they use this bit.

Nanoseconds: 30 bits The nanoseconds field contains the number of nanoseconds of the timestamp. When a system is not capable of providing timestamps with nanosecond resolution, the highest resolution that can be provided is used and the other digits of the nanosecond field are set to 0. When a system uses, e.g., a 16-bit subdivision of one second (such as NTP timestamps may), the decimal value is rounded to the nearest nanosecond. Adding or subtracting a certain random number of nanoseconds to the result may be used to prevent rounding bias. This is up to the implementer of the IP measurement option for a system, but must be documented.

Unique Identifier (UID): 32 bits The Unique Identifier is incremented for each sent IP packet. This allows the receiving host to identify packet losses, duplications and re-orderings. It must be noted that each fragment of a fragmented datagram contains a copy of the IP measurement option. Therefore systems "on the path" should rely, e.g., on the fragment offset to perform such measurements. End-hosts perform measurements on the packet before fragmenting and after re-assembly. Note that the 16 least significant bits of the UID can be used with the IPv4 algorithms, since they roll over in similar fashion. The full 32-bit UID has relevance on very fast IPv6 links.

(Optional) Cryptographic Signature: Optionally up to 63 words, 252 octets, or 2016 bits can be used for the transmission of a cryptographic signature.

As the flow identifier is an integral 20-bit field in the IPv6 header it is not included in the IPv6 measurement option. When the option is encrypted only the first two octets have their normal meaning. The rest of the octets form the cyphertext.

This option shall be aligned in a 4n fashion.

9. Using the IP measurement option

The IP MO was defined to enable the passive measurement of various statistics on Diffserv microflows [RFC2474]. Amongst others these are: - One-way Packet Delay, Mean Packet Delay, Packet Delay Variation and related statistics. - Packet Losses, Reordering and Duplication

The first goal is to measure such statistics on microflows [RFC2474] between two end hosts at a point-to-point (P2P) IP link. The statistics can of course be extended to multi-point-to-multi-point (MP2MP) links. This requires that the packets can be uniquely assigned to a microflow at the IP layer and that all properties needed to perform the measurements must be available either by a statistics collector or by the destination host(s). It is assumed that each microflow is assigned a unique flow identifier by the source host. This means that each connection has two microflows, each with its own flow identifier.

A second goal is that hosts en route, can try to measure or estimate the measurands on a microflow basis. The difficulty here is that not all packets must pass a certain host on the route. Nevertheless, useful information can be obtained from the estimations made by on route hosts. E.g. the fraction of the traffic traversing certain routes A, B, or C.

As explained, timestamps upon transmission and reception are needed to perform delay measurements and their related measurants. To perform packet loss and related measurements an incrementing numeration counter is needed (NUM).

9.1. Measurement procedure on the end hosts

It is more or less assumed that microflows belong to a pair of sockets on the source and destination hosts. The operating system (OS) makes counters available that can be used by polling or (when a fancier way is needed) by an interrupt mechanism. Such interrupts can come at a programmable time, e.g. every second. This time interval will be termed measurement interval (MI). Since the OS has the timers, this seems a logical scheme. The presence of the MO may trigger the calculation of the various statistics, this can be a configuration option.

Hosts can make two queues available to user space programs where the outgoing and incoming IP headers of each flow are queued for user space analysis. For incoming packets the reception timestamp is also inserted in this queue.

It must however be considered that the statistics are only reliable after the maximum packet delay (MPD) of the system, known to the operator, has expired. This MPD must thus be configurable but may be set to a default value of 120 seconds. It is advised to make these settings configurable for per microflow at the destination host. It is advised that each MI becoming available after MPD is tagged with the start time of the interval at the TAI timescale [IEEE1588-2019] also used in the MO tagged packets.

Delay and its related metrics can be readily determined when the destination host knows the time when the packet was sent. The source time stamp is included in the MO option. If no packets are lost delay statistics can be presented immediately to the user.

Losses can be calculated when the packet counters on the source and destination are compared, but also when the destination knows which packets were lost. The packet number (NUM) is therefore included in the MO together with the time stamp.

When an IP packet is created the space for an MO is reserved in the packet’s option header and is indicated with a pointer in memory. The flow label can be inserted readily. Just before transmission of the first octet of the IP packet a timestamp is retrieved from the OS kernel (transmission time stamp TTS) and inserted into the IP packet. The colouring counter NUM is inserted and increased. The packet is then transmitted by the OS. After transmission the kernel updates the kernel counters for transmission. It may, but must not, make the IP header available for analysis by forwarding it to a queue that can be read by user space programs.

The receiving host has memory reserved for a timestamp and the data of an incoming IP packet header. When the receiving host sees that an IP packet carries the MO option, it generates a timestamp (reception time stamp RTS) when the last octet of the packet has been received. When the completeness of the packet has been verified (checksums), the MO data and the RTS are used to update the kernel statistics. The system may also make the IP header and the RTS available for analysis by forwarding it to a queue that can be read by user space programs.

At the receiving host the following data are used to indicate the flow: - source IP address - destination IP address - flow ID

Normally a microflow would be known based on the protocol ID and source and destination ports, these can still be used for backward compatibility or to calculate metrics on microflows that do not carry the MO option. Thoughts on raw data formats for statistical analysis by a data collection system are given in Notes for raw metering protocols.

9.2. Measurement procedure at observers

Each switch with deep packet inspection capabilities, each router and host observing IP packets with an MO option can generate a RTS upon complete reception. Statistics can now be calculated for each microflow. However, care must be taken when interpreting these statistics, but they may be useful for network fault analysis. In the remainder of this section the possibilities and limitations for measurements made by observers will be discussed.

The observation on the MPD for end hosts, also holds for observing hosts. Therefore, the administrator must be able to set the MPD for gathering statistics. An observer may make raw packet headers available for analysis to other systems. Observers may contain lists that limit the number of hosts for which to gather statistics. This is not very different from the way Diffserv determines which microflows to process or not.

A speciality for observers is that they cannot be certain that all packets that are not lost, pass the system. Therefore, care must be taken when calculating statistics. For example, a loss statistic may say more about the fraction of microflow packets passing a system than on the actual loss. With timestamps and an ID, such fractions can even be detected with high certainty.

By means of the MO an observer can calculate all statistics that a destination host can. It would be possible to introduce additional metrics that are specifically tuned for observers. For example, metrics that only take largely complete sets of packets to calculate losses, reordering and duplication.

The following properties can at least be observed with the standard metrics: - Fraction of packets passing the observer. - One-way Delay and related statistics from source to observer. - Packet reordering and duplicates on consecutive sequences of packets.

For a full set of observers, one could extend the measurement scheme by sending the MO and observer RTS to an external measurement system. Such an external system can then calculate more than the individual host. The raw data that these systems must transmit to enable this are those the destination host transmits.

9.3. Metrics and Measurement protocols

The NUM+TTS information is used to generate statistics on the One-way packet loss, the one-way packet reordering and the one-way packet duplication. Metrics for these properties can be related to those available in the IP Performance Metrics (IPPM) framework [RFC2330], [RFC7680]. The same holds for Delay based metrics [RFC7679], [RFC3393]. On the other hand, the metrics could be taken from the Ethernet specifications [MEF10.4]. This framework seems to be more mature at the moment.

When the destination host makes the MO option and necessary information for metric determination available to the user space. Advanced metric calculations must not take place in kernel space and user space programs can be used to make measurement data available according to one or more sets of metrics, or may even send the raw data off to a storage server for later processing. The idea that information can be available in real-time must not be followed since the MPD must always be observed.

To send measurement statistics to an overarching data collection system, protocols like RMON [RFC2819] and its extensions or IPFIX [RFC7011] can be used or extended.

9.4. Notes for systems with real-time protocols

For systems relying on real-time protocols in the network an MPD of 120 seconds, or even 10 seconds, may be way to long. Depending on the criticality of knowing the network performance near real time, the MPD on sockets involved, can be set much lower, e.g. 1 second. This is based on the fact that packets that have longer delay have no meaning anymore for such protocols and are deemed lost.

9.5. Notes for raw metering data protocols

When sending raw data to an analysis system for full system analysis the following must be included in a data unit per package. Such a feature can, of course, only be used when the network has sufficient capacity left to send this additional data. A protocol specification is left to the implementer.

One data unit would contain a flow identification [RFC6437] consisting of: - source IP address (32 or 128 bit) - destination IP address (32 or 128 bit) - flow ID (20 bit) or, when it occupies less bits, a hash (e.g. for IPv6), and measurement data: - NUM field (16 bit) - TTS nanosecond counter (32 bit) - TTS second counter (12 bit) - RTS nanosecond counter (32 bit) - RTS second counter (12 bit)

for IPv4 this is a total of 188 bits. For IPv6 this could be 160-bit for a flow hash (e.g. SHA1) and 104-bit for the measurement data (264 bit). It is proposed that source hosts set the RTS fields to 0.

Multiple data units can be packed in one IP packet for the measurement system A specification for such a protocol can be created when needed. This could include the traditional Diffserv ports + protocol for IPv4 for non-MO enabled measurements.

A consideration could be to include in these data the following data for the observer transmission time: - ONUM field (16 bit) - OTTS nanosecond counter (32 bit) - OTTS second counter (12 bit) where ONUM is the observer NUM field at transmission, and OTTS is the observer transmission time stamp. This would allow to separate wire and system time for an observer.

Alternatively, packets can almost always be identified based on a hash. Hash based measurement methods [RFC7014] would nevertheless need to send at least the timestamps in order to determine the system delays for packets. It is therefore not necessarily cheaper (bit and computation wise) to use a hash based measurement approach.

10. Security Considerations

Security measures were considered during the design of the IP option, but, especially in the case of the IPv4 measurement option, possibilities are limited due to the limited available space in the IPv4 header. Nevertheless, dropping of the IP measurement option cannot be prevented by the end-hosts.

As an alternative to real security measures, it is also possible to share information on how to obfuscate the measurement option information for third parties. The timestamps and NUM fields may be manipulated by a PRNG of sufficient strength to make systems on the packet path that are not in the owner’s hand, guess about the true values. The systems owned by the entity doing the measurements, can share the seeds of the PRNGs, and regularly update these, using out of band securely encrypted channels.

It must be considered that, with current network systems, data can be recorded in abundance, and that obfuscating algorithms may be broken quickly, allowing the third party to at least read the data in the IP measurement option.

However, the author of this RFC would not endorse such a method as secure.

11. IANA Considerations

This document requests two IP option numbers to be registered by IANA.

12. Conclusion

An IPv4 and IPv6 option was introduced that is flexible enough to support various measurement schemes on various types of IP traffic. Although it ads to the length of the IP packet, it can be used to measure traffic in situations where full statistics are needed (e.g., systems with stringent SLS requirements on delay and packet loss) and bandwidth is not the limitation for applications.

Measurement procedures and relevant metrics are to be discussed in companion RFCs.

13. Disclaimer and Patents

Parts of this RFC contain information about technology that is patented. Identified patents at Siemens Mobility are: * WO 2023/030908 Al; DE 10 2021 209 622 A1; Verfahren zum Betreiben einer Eisenbahngleisanlage

Related patents at Siemens Mobility are: * DE102024204819A1; EP4654562A1; Verfahren zum Erweitern eines Headers eines Datenpakets Further patents are pending.

14. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

Acknowledgments

Tjeerd Pinkert wants to thank Gert Bolz, and Benjamin Schilling for their support of the passive network measurements innovation project, and Sascha Liebscher, Achim Willers, Tobias Grosch, and Jaime Lazaro Calderon for their support to make this work possible within Siemens Mobility.

Contributors

Benjamin Schilling
Siemens Mobility GmbH
Ackerstrasse 22
38126 Braunschweig
Germany
Email: benjamin.schilling@siemens.com
URI: https://www.mobility.siemens.com

Gert Bolz
Siemens Mobility GmbH
Ackerstrasse 22
38126 Braunschweig
Germany
Email: gert.bolz@siemens.com
URI: https://www.mobility.siemens.com

Authors' Addresses

dr. ir. Tjeerd J. Pinkert
Siemens Mobility GmbH
Ackerstrasse 22
38126 Braunschweig
Germany
Email: tjeerd.pinkert@siemens.com
URI: https://www.mobility.siemens.com

Negar Masoudifar
Siemens Mobility GmbH
Ackerstrasse 22
38126 Braunschweig
Germany
Email: negar.masoudifar@siemens.com
URI: https://www.mobility.siemens.com

Akachukwu Adnife Nnamdi
Siemens Mobility GmbH
Ackerstrasse 22
38126 Braunschweig
Germany
Email: akachukwu.nnamdi@siemens.com
URI: https://www.mobility.siemens.com