ippm
Internet-Draft
Intended status: Standards Track
Expires: 26 February 2026

J. Iurman
University of Liege
T. Zhou
Huawei

25 August 2025

A YANG Data Model for In Situ Operations, Administration, and Maintenance (IOAM) Integrity Protected Options
draft-iuzh-ippm-ioam-integrity-yang-02

Abstract

In Situ Operations, Administration, and Maintenance (IOAM) is an example of an on-path hybrid measurement method. IOAM defines a method for producing operational and telemetry information that may be exported using the in-band or out-of-band method. I-D.ietf-ippm-ioam-data-integrity (RFC Ed.: to be replaced by RFC YYYY) defines IOAM Options with integrity protection, also called Integrity Protected Options. This document defines a YANG module for the configuration of these Integirty Protected Options.

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/.

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 26 February 2026.

Copyright Notice

Table of Contents

1. Introduction

In Situ Operations, Administration, and Maintenance (IOAM) is an example of an on-path hybrid measurement method. IOAM defines a method for producing operational and telemetry information that may be exported using the in-band or out-of-band method. [I-D.ietf-ippm-ioam-data-integrity] defines IOAM Options with integrity protection, also called Integrity Protected Options. This document defines a data model for the configuration of these Integirty Protected Options using the YANG data modeling language [RFC7950]. This YANG data model supports four IOAM Integrity Protected Options, which are as follows:

* Integrity Protected Incremental Trace-Option [I-D.ietf-ippm-ioam-data-integrity]
* Integrity Protected Pre-allocated Trace-Option [I-D.ietf-ippm-ioam-data-integrity]
* Integrity Protected Proof of Transit (POT) Option [I-D.ietf-ippm-ioam-data-integrity]
* Integrity Protected Edge-to-Edge (E2E) Option [I-D.ietf-ippm-ioam-data-integrity]

2. Conventions

2.1. Requirements Language

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.

2.2. Abbreviations

Abbreviations used in this document:

OAM: Operations, Administration, and Maintenance
IOAM: In Situ OAM
POT: Proof of Transit
E2E: Edge to Edge

2.3. Terminology

The following terms are defined in [RFC7950] and are used in this specification:

* augment
* data model
* data node

The terminology for describing YANG data models is found in [RFC7950].

2.4. Tree Diagrams

Tree diagrams used in this document follow the notation defined in [RFC8340].

3. Design of the IOAM Integrity YANG Data Model

3.1. Overview

The IOAM Integrity model is organized as a list of profiles, as shown in the following figure.

```
module: ietf-ioam-integrity

  augment /ioam:ioam/ioam:profiles/ioam:profile:
    +--rw int-incremental-tracing-profile! {int-incremental-trace}?
    |  ...
    +--rw int-preallocated-tracing-profile! {int-preallocated-trace}?
    |  ...
    +--rw int-pot-profile! {int-proof-of-transit}?
    |  ...
    +--rw int-e2e-profile! {int-edge-to-edge}?
```

This document uses the "ietf-ioam" YANG module [RFC9617] and augments its definition of a profile. The supported profiles are indicated by four defined features, i.e., "int-incremental-trace", "int-preallocated-trace", "int-proof-of-transit", and "int-edge-to-edge" (i.e., "int" prefix for "INTegrity protection"). Although these four new profiles resemble those defined in [RFC9617], they are distinct profiles since they represent different IOAM Option-Type code points.

The YANG data model in this document conforms to the Network Management Datastore Architecture (NMDA) defined in [RFC8342].

3.2. Integrity Protected Pre-allocated Tracing Profile

The "int-preallocated-tracing-profile" parameter contains the detailed information for the pre-allocated tracing data with integrity protection. This information is the same as for the Pre-allocated Tracing Profile; see [RFC9617], Sec. 3.2. This information also includes:

int-method: indicates which Integrity Protection Method is used.

```
+--rw int-preallocated-tracing-profile! {int-preallocated-trace}?
   +--rw node-action?   ioam-node-action
   +--rw trace-types
   |  +--rw use-namespace?   ioam-namespace
   |  +--rw trace-type*      ioam-trace-type
   +--rw max-length?    uint32
   +--rw int-method?    method-type
```

3.3. Integrity Protected Incremental Tracing Profile

The "int-incremental-tracing-profile" parameter contains the detailed information for the incremental tracing data with integrity protection. This information is the same as for the Integrity Protected Pre-allocated Tracing Profile; see Section 3.2.

```
+--rw int-incremental-tracing-profile! {int-incremental-trace}?
   +--rw node-action?   ioam-node-action
   +--rw trace-types
   |  +--rw use-namespace?   ioam-namespace
   |  +--rw trace-type*      ioam-trace-type
   +--rw max-length?    uint32
   +--rw int-method?    method-type
```

3.4. Integrity Protected Proof of Transit Profile

The "int-pot-profile" parameter is intended to contain the detailed information for the proof of transit data with integrity protection. This information is the same as for the Proof of Transit Profile; see [RFC9617], Sec. 3.5. This information also includes:

node-action: the same semantic as provided in Section 3.2.
int-method: the same semantic as provided in Section 3.2.

```
+--rw int-pot-profile! {int-proof-of-transit}?
   +--rw use-namespace?   ioam:ioam-namespace
   +--rw pot-type?        ioam:ioam-pot-type
   +--rw node-action?     ioam:ioam-node-action
   +--rw int-method?      method-type
```

3.5. Integrity Protected Edge-to-Edge Profile

The "int-e2e-profile" parameter contains the detailed information for the edge-to-edge data with integrity protection. This information is the same as for the Edge-to-Edge Profile; see [RFC9617], Sec. 3.6. This information also includes:

int-method: the same semantic as provided in Section 3.2.

```
+--rw int-e2e-profile! {int-edge-to-edge}?
   +--rw node-action?   ioam-node-action
   +--rw e2e-types
   |  +--rw use-namespace?   ioam-namespace
   |  +--rw e2e-type*        ioam-e2e-type
   +--rw int-method?    method-type
```

4. IOAM Integrity YANG Module

The "ietf-ioam-integrity" module defined in this document imports the "ietf-ioam" module defined in [RFC9617]. This document also references [I-D.ietf-ippm-ioam-data-integrity].

<CODE BEGINS> file "ietf-ioam-integrity@2025-08-25.yang"

```
module ietf-ioam-integrity {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-ioam-integrity";
  prefix "ioam-int";

  import ietf-ioam {
    prefix ioam;
    reference
      "RFC 9617: A YANG Data Model for In Situ Operations,
       Administration, and Maintenance (IOAM)";
  }

  organization
    "IETF IPPM (IP Performance Measurement) Working Group";

  contact
    "WG Web:   <https://datatracker.ietf.org/wg/ippm>
     WG List:  <mailto:ippm@ietf.org>
     Author:   Tianran Zhou
               <mailto:zhoutianran@huawei.com>
     Author:   Justin Iurman
               <mailto:justin.iurman@uliege.be>";

  description
    "This YANG module specifies a vendor-independent data model for
     In Situ Operations, Administration, and Maintenance (IOAM)
     Integrity Protected Options.

     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 (RFC 2119) (RFC 8174) when, and only when,
     they appear in all capitals, as shown here.

     Copyright (c) 2025 IETF Trust and the persons identified as
     authors of the code.  All rights reserved.

     Redistribution and use in source and binary forms, with or
     without modification, is permitted pursuant to, and subject to
     the license terms contained in, the Revised BSD License set
     forth in Section 4.c of the IETF Trust's Legal Provisions
```

```
        Relating to IETF Documents
        (https://trustee.ietf.org/license-info).

        This version of this YANG module is part of RFC XXXX; see the
        RFC itself for full legal notices.";

      revision 2025-08-25 {
        description
          "Initial revision.";
        reference
          "RFC XXXX: A YANG Data Model for In Situ Operations,
           Administration, and Maintenance (IOAM) Integrity Protected
           Options";
      }

      /*
       * FEATURES
       */

      feature int-incremental-trace
      {
        description
          "This feature indicates that the Integrity Protected
           Incremental Trace-Option is supported.";
        reference
          "RFC YYYY: Integrity Protection of In Situ Operations,
           Administration, and Maintenance (IOAM) Data Fields";
      }

      feature int-preallocated-trace
      {
        description
          "This feature indicates that the Integrity Protected
           Pre-allocated Trace-Option is supported.";
        reference
          "RFC YYYY: Integrity Protection of In Situ Operations,
           Administration, and Maintenance (IOAM) Data Fields";
      }

      feature int-proof-of-transit
      {
        description
          "This feature indicates that the Integrity Protected Proof of
           Transit Option is supported.";
        reference
          "RFC YYYY: Integrity Protection of In Situ Operations,
           Administration, and Maintenance (IOAM) Data Fields";
      }
```

```
  feature int-edge-to-edge
  {
    description
      "This feature indicates that the Integrity Protected
       Edge-to-Edge Option is supported.";
    reference
      "RFC YYYY: Integrity Protection of In Situ Operations,
       Administration, and Maintenance (IOAM) Data Fields";
  }

  /*
   * IDENTITIES
   */

  identity method {
    description
      "Base identity to represent the Integrity Protection Method.";
  }

  identity method-1 {
    base method;
    description
      "The Integrity Protection Method 1 uses AES-GMAC with a 12-byte
       Nonce and a 16-byte ICV.";
    reference
      "RFC YYYY: Integrity Protection of In Situ Operations,
       Administration, and Maintenance (IOAM) Data Fields";
  }

  /*
   * TYPE DEFINITIONS
   */

  typedef method-type {
    type identityref {
      base method;
    }
    description
      "It specifies the Integrity Protection Method.";
  }

  /*
   * DATA NODES
   */

  augment "/ioam:ioam/ioam:profiles/ioam:profile" {
    description
      "This augmentation adds 4 profiles for the Integrity Protected
```

```
          Options.";

      container int-incremental-tracing-profile {
        if-feature "int-incremental-trace";
        presence
          "Enables the Integrity Protected Incremental Trace-Option.";
        description
          "This container describes the profile for the Integrity
           Protected Incremental Trace-Option.";

        uses ioam:ioam-incremental-tracing-profile;

        leaf int-method {
          when "derived-from-or-self(../node-action,
             'ioam:action-encapsulate')";
          type method-type;
          default "method-1";
          description
            "This object indicates the Integrity Protection Method for
             this profile.";
        }
      }

      container int-preallocated-tracing-profile {
        if-feature "int-preallocated-trace";
        presence
          "Enables the Integrity Protected Pre-allocated
           Trace-Option.";
        description
          "This container describes the profile for the Integrity
           Protected Pre-allocated Trace-Option.";

        uses ioam:ioam-preallocated-tracing-profile;

        leaf int-method {
          when "derived-from-or-self(../node-action,
             'ioam:action-encapsulate')";
          type method-type;
          default "method-1";
          description
            "This object indicates the Integrity Protection Method for
             this profile.";
        }
      }

      container int-pot-profile {
        if-feature "int-proof-of-transit";
        presence
```

```
          "Enables the Integrity Protected Proof of Transit Option.";
        description
          "This container describes the profile for the Integrity
           Protected Proof of Transit Option.";

        leaf use-namespace {
          type ioam:ioam-namespace;
          default "ioam:default-namespace";
          description
            "This object indicates the namespace used for the
             POT types.";
        }

        leaf pot-type {
          type ioam:ioam-pot-type;
          description
            "The type of a particular POT variant that specifies
             the POT data that is included.";
        }

        leaf node-action {
          type ioam:ioam-node-action;
          default "ioam:action-transit";
          description
            "This object indicates the action the node needs to
             take, e.g., encapsulation.";
        }

        leaf int-method {
          when "derived-from-or-self(../node-action,
             'ioam:action-encapsulate')";
          type method-type;
          default "method-1";
          description
            "This object indicates the Integrity Protection Method for
             this profile.";
        }
      }

      container int-e2e-profile {
        if-feature "int-edge-to-edge";
        presence
          "Enables the Integrity Protected Edge-to-Edge Option.";
        description
          "This container describes the profile for the Integrity
           Protected Edge-to-Edge Option.";

        uses ioam:ioam-e2e-profile;
```

```
        leaf int-method {
          when "derived-from-or-self(../node-action,
             'ioam:action-encapsulate')";
          type method-type;
          default "method-1";
          description
            "This object indicates the Integrity Protection Method for
             this profile.";
        }
      }
    }
  }
  <CODE ENDS>
```

5. Security Considerations

Security considerations listed in [RFC9617] also apply here. This document does not introduce new security considerations.

6. IANA Considerations

RFC Ed.: In this section and in Section 4, please replace all occurrences of 'RFC XXXX' with the actual RFC number. Also in Section 4 and in the Abstract, please replace all occurrences of 'RFC YYYY' with the actual RFC number of [I-D.ietf-ippm-ioam-data-integrity] (and remove this note).

IANA is requested to assign a new URI from the "IETF XML Registry" [RFC3688]. The following URI is suggested:

URI: urn:ietf:params:xml:ns:yang:ietf-ioam-integrity
Registrant Contact: The IESG.
XML: N/A; the requested URI is an XML namespace.

This document also requests a new YANG module name in the "YANG Module Names" registry [RFC6020] with the following suggestion:

Name: ietf-ioam-integrity
Namespace: urn:ietf:params:xml:ns:yang:ietf-ioam-integrity
Prefix: ioam-int
Reference: RFC XXXX

7. Acknowledgements

The authors would like to thank Alex Huang Feng for his feedback.

8. Normative References

[I-D.ietf-ippm-ioam-data-integrity]
Brockners, F., Bhandari, S., Mizrahi, T., and J. Iurman, "Integrity Protection of In Situ Operations, Administration, and Maintenance (IOAM) Data Fields", Work in Progress, Internet-Draft, draft-ietf-ippm-ioam-data-integrity-14, 25 August 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-ippm-ioam-data-integrity-14>.

[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/info/rfc2119>.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <https://www.rfc-editor.org/info/rfc3688>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/info/rfc6020>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[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/info/rfc8174>.

[RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <https://www.rfc-editor.org/info/rfc8340>.

[RFC8342] Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "Network Management Datastore Architecture (NMDA)", RFC 8342, DOI 10.17487/RFC8342, March 2018, <https://www.rfc-editor.org/info/rfc8342>.

[RFC9617] Zhou, T., Ed., Guichard, J., Brockners, F., and S. Raghavan, "A YANG Data Model for In Situ Operations, Administration, and Maintenance (IOAM)", RFC 9617, DOI 10.17487/RFC9617, August 2024, <https://www.rfc-editor.org/info/rfc9617>.

Appendix A. An Example of the Integrity Protected Pre-allocated Tracing Profile

TBD

Appendix B. An Example of the Integrity Protected Incremental Tracing Profile

TBD

Appendix C. An Example of the Integrity Protected Proof of Transit Profile

TBD

Appendix D. An Example of the Integrity Protected Edge-to-Edge Profile

TBD

Authors' Addresses

Justin Iurman
University of Liege
10, Allee de la decouverte (B28)
4000 Sart-Tilman
Belgium
Email: justin.iurman@uliege.be

Tianran Zhou
Huawei
156 Beiqing Rd.
Beijing
100095
China
Email: zhoutianran@huawei.com