

# Integrity of In-situ OAM Data Fields

[draft-ietf-ippm-ioam-data-integrity-09](#)

Justin Iurman, Frank Brockners, Shwetha Bhandari, Tal Mizrahi

IETF 120, IPPM WG  
July 23, 2024

# Status -09

- Addresses Ben's review (early secdir)
- Includes feedback received in Brisbane
- Document ready for WGLC
- ... but, before that, three questions to the WG (see next slides)

# Question 1/3: Nonce (*RECOMMENDED* vs *MUST*)

Context: specific size for the Nonce, with specific content

Advantage of *RECOMMENDED*:

- not restrictive

Advantage of *MUST*:

- security: optimized key usage (based on NIST recommendations)
- interop: encapsulating node identification, replay attack protection

Note: -09 uses *MUST*... Everyone OK with this change?

# Question 2/3: Unknown Integrity Protection Method

Method-ID: 8-bit unsigned integer. It defines the Integrity Protection Method to compute the Integrity Check Value (ICV) field. If a node encounters an unknown value, it MUST NOT change the contents of the IOAM Integrity Protection header and MUST NOT change the contents of the IOAM-Data-Fields. In other words, the node MUST NOT process the IOAM Option-Type.

← current version in -09

... a node does NOT insert IOAM-Data-Fields (IOAM would not “work” on the node anymore)

vs.

Method-ID: 8-bit unsigned integer. It defines the Integrity Protection Method to compute the Integrity Check Value (ICV) field. If a node encounters an unknown value, it MUST NOT change the contents of the IOAM Integrity Protection header.

... a node inserts IOAM-Data-Fields anyway (IOAM “works” but also breaks the Integrity validation)

Any opinions?

## Question 3/3: Ambiguity of “(im)mutable” with pre-alloc

Whatever the Option-Type, a transit node MUST NOT participate in the integrity protection (i.e., update the ICV) if it does not add IOAM-Data-Fields → **mandatory** to not break the integrity validation.

### Text attempt:

*“If the transit node does not add any IOAM-Data-Fields (e.g., it only modifies mutable IOAM-Data-Fields or does nothing), then the transit node MUST NOT update the ICV field in the IOAM Integrity Protection header.”*

... would not work for the Pre-allocated Trace: they don't really “add” IOAM-Data-Fields, instead they “update” or “modifies” IOAM-Data-Fields (pre-allocated by the encapsulating node) → they are considered more as mutable rather than immutable fields in this case.

How to distinguish pre-alloc from others? Working on it, any suggestions appreciated.