Integrity of In-situ OAM Data Fields
In-situ OAM Deployment

draft-ietf-ippm-ioam-data-integrity-00
draft-ietf-ippm-ioam-deployment-00

IETF 112, IPPM WG
November 11, 2021

Integrity of In-situ OAM Data Fields

[draft-ietf-ippm-ioam-data-integrity-00]

Frank Brockners, Shwetha Bhandari, Tal Mizrahi

IETF 112, IPPM WG
November 11, 2021

Status, Updates

- draft-brockners-ippm-ioam-data-integrity-02 WG adopted

- draft-ietf-ippm-ioam-data-integrity-00 published with updates
  - Information -> Standards track
    - Evolved document from a “document which discusses options” to a “specification”: Removed paragraphs which related to the discussion of solution approaches
    - Removed appendix A which listed solution approaches which were not chosen by the WG (i.e., “Method 1”, “Method 2”, “Method 4”, “Method 5”)
  - Editorial
    - Added note that the draft is to protect all IOAM Option-Types (incl. e.g., DEX) and that the IOAM-Trace Option-Type is only used to illustrate the methods.
    - Alignment with draft-ietf-ippm-ioam-data nomenclature (“intermediate nodes” -> “IOAM transit nodes”)
    - Wording changes, nit fixes to improve readability

Next Steps

- Expand/add “Security Considerations” section (section 6)
- Garner additional input on early candidates to use for digest and signature algorithms (see section 5.2 IOAM Integrity Protection Algorithm Suite Registry)
- Further reviews

In-situ OAM Deployment

draft-ietf-ippm-ioam-deployment-00

Frank Brockners, Shwetha Bhandari, Daniel Bernier, Tal Mizrahi

IETF 112, IPPM WG
November 11, 2021

Status, Updates

- draft-brockners-opsawg-ioam-deployment-03 WG adopted

- draft-ietf-ippm-ioam-deployment-00 published with updates
    - Alignment with RFC 8799
      (-> “limited domains”)
    - Alignment with draft-ietf-ippm-ioam-data nomenclature
      (“source/sink” -> “IOAM encap/decap nodes”)
    - Wording changes to improve readability

Next Steps

- Continue to include deployment experiences;
  - IOAM Implementation in Linux Kernel (version 5.15) and FD.io/VPP continue to evolve*
  - Wireshark starts to support IOAM (currently: Pre-allocated Trace Option-Type)*
- Further reviews

* See also: https://github.com/Advanced-Observability