WG Working Group
Internet-Draft
Intended status: Informational
Expires: 19 June 2023

S. Salsano
Univ. of Rome Tor Vergata / CNIT
H. ElBakoury
Consultant
D. Lopez
Telefonica, I+D

16 December 2022

Extensible In-band Processing (EIP) Architecture and Framework

draft-eip-arch-01

Abstract

Extensible In-band Processing (EIP) ~~extends the functionality of the~~is an IPv6 extension that is designed to cover ~~protocol considering~~ the needs of ~~future Internet~~novel ~~-~~services ~~/ 6G networks~~. This document discusses the overall ~~architecture~~ architectural ~~and~~ framework of EIP. Two companion ~~separate~~ documents ~~respectively~~ are edited to analyze a number of use cases for EIP and provide the protocol specifications of EIP.

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://eip-home.github.io/eip-headers/draft-eip-arch.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-eip-arch/.

Discussion of this document takes place on the EIP SIG mailing list (mailto:eip@cnit.it), which is archived at http://postino.cnit.it/cgi-bin/mailman/private/eip/.

Source for this draft and an issue tracker can be found at https://github.com/eip-home/eip-arch.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Networking architectures need to evolve to support the needs of future ~~Internet~~ services, e.g., those that are envisaged for B5G and ~~and~~ 6G networks. The networking research and standardization communities have considered different approaches for this evolution, that can be broadly classified in three~~3~~ different categories:

Commenté [BMI1]: You may include what is specific to these "services" and why conventional tools are not sufficient. Examples would be useful for the reader.

Commenté [BMI2]: This is not specific to 6G, IMO.

1. Clean slate ~~and~~ (a.k.a., "revolutionary") solutions. Throw away the legacy IP networking layer.

2. Solutions above ~~the~~ layer 3~~.~~: Do not ~~touch~~ require any change to the legacy networking layer (IP).

3. Evolutionary solutions~~.~~: Improve the IP layer (and try to preserve backward compatibility).

The proposed EIP (Extensible In-band Processing) solution belongs to the third category as~~,~~ it extends the ~~current~~ IPv6 architecture without requiring a clean-slate revolution.

The use cases for EIP are discussed in [id-eip-use-cases]. The specification of the EIP header format is provided in [id-eip-headers].

5. 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.

2. Basic ~~principles~~ Principles for EIP

An ongoing trend is to extend~~ing~~ the functionality of the IPv6 networking layer, going beyond the plain packet forwarding. An example of this trend is ~~the rise of~~ the SRv6 "network programming" model. With ~~the~~ such a ~~SRv6 network~~ programming model, ~~the~~ routers can implement "complex" functionalities and they can be controlled by a "network program" that is embedded in IPv6 packet headers. Another example is the INT (IN-band Telemetry) solution for monitoring. These (and other) examples are further discussed in Section 4.

Commenté [BMI3]: That is ?

The EIP solution is aligned with this trend, which will ensure a ~~future~~ future-proof evolution of networking architectures. EIP supports a feature-rich and extensible IPv6 networking layer, in which complex dataplane functions can be executed by end-hosts, routers, virtual functions, servers in datacenters so that services can be implemented in the smartest and more efficient way.

Commenté [BMI4]: Rather than using this language, I suggest you sue a more factual wording by calling out what is meant here

Commenté [BMI5]: Not sure what is meant here.

The EIP solution foresees the introduction of an EIP IPv6 extension header ~~in the IPv6 packet header~~. The proposed EIP header is extensible and it is meant to support a number of different use cases. In general, both end-hosts and transit routers can read and write the content of this header. Depending of the specific use-case, only specific nodes will be capable and interested in reading or writing the EIP header. The use of the EIP header can be confined to a single domain or to a set of cooperating domains, so there is no need of a global, Internet-wide support of the new header for its introduction. Moreover, there can be ~~usage~~ scenarios in which legacy nodes can simply ignore the EIP header and provide transit to packets containing the EIP header.

Commenté [BMI6]: How this is signaled to these intermediate routers?

Commenté [BMI7]: Please add a pointer to the base IPv6 to back this claim.

An important usage scenario considers the transport of user packets over a provider network. In this scenario, we consider the network portion from the provider ingress edge node to the provider egress edge node. The ingress edge node can encapsulate the user packet coming from an access network into an outer packet. The outer packet travels in the provider network until the egress edge node, which will decapsulate the inner packet and deliver it to the destination access network or to another transit network, depending on the specific topology and service. Assuming that the IPv6/SRv6 data plane is used in the provider network, the ingress edge node will be the source of an outer IPv6 packet in which it is possible to add the EIP header. The outer IPv6 packet (containing the EIP header) will be processed inside the "limited domain" (see [RFC8799]) of the provider network, so that the operator can make sure that all the transit routers either are EIP aware or at least they can forward packets containing the EIP header. In this usage scenario, the EIP framework operates "edge-to-edge" and the end-user packets are "tunneled" over the EIP domain.

Commenté [BMI8]: Hmm, isn't this what normal IP transfer capability is about 😊

Commenté [BMI9]: Unless the host supports EIP, this is the only "allowed" usage given that is not allowed inject EHs without encapsulation.

The architectural framework for EIP is depicted in Figure 1. We refer to the nodes that are not ~~EIP~~ EIP-capable as legacy nodes. An EIP domain is made up by ~~EIP~~ EIP-aware routers (EIP R) and can also include legacy routers (LEG R). At the border of the EIP domain, EIP edge nodes (EIP ER) are used to interact with legacy End Hosts / Servers (LEG H) and with other domains. It is also possible that an End Host / Server is ~~EIP~~ EIP-aware (EIP H), in this case, the EIP ~~framework~~ could operate "edge-to-end" or "end-to-end".

Commenté [BMI10]: This text can be positioned before the edge-to-edge discussion.

```
                                                 LEG domain
                                               +-------------+

     +----+              +----+     +----+                +----+
     |EIP|_             _|EIP|______|EIP|              ___|LEG|
     | H | \__+----+__/ | R |      | R |__    +----+__/   | R | ...
     +----+    |EIP|     +----+     +----+  \__|EIP|      +----+
             __|ER |__      |          |      __|ER |__
     +----+_/  +----+  \_+----+     +----+__/  +----+  \___+----+
     |LEG|               |LEG|______|LEG|                |EIP|
     | H |               | R |      | R |                |ER | ...
     +----+              +----+     +----+                +----+

                +------------------------------+         +-------------+
                          EIP domain                        EIP domain
```

Figure 1: EIP ~~framwork~~Framework

As shown in Figure 1, an EIP domain can communicate with other domains, which can be legacy domains or EIP capable domains.

Commenté [BMI11]: This may be perceived as conflicting with “limited domain” assumption.

3. Benefits of a common EIP header for multiple use cases~~.~~

Commenté [BMI12]: Rather than including this section, it would be useful to discuss why another EIP extension is needed, rather than leveraging SRH, for example.

The EIP header will carry different EIP Information Elements that are defined to support the different use cases. There are reasons why it is beneficial to define a common EIP header that supports multiple use cases.

Commenté [BMI13]: Would be useful to provide some examples.

1. The number of available Option Types in HBH header is limited, likewise the number of available TLVs in the Segment Routing Header (SRH) is limited. Defining multiple Option Types or SRH TLVs for multiple use case is not scalable and puts pressure on the allocation of such codepoints. This aspect is further discussed in Section 4.

2. The definition and standardization of specific EIP Information Elements for the different use cases will be simplified, compared to the need of requiring the definition of a new Option Type or SRH TLVs.

3. Different use cases may share a subset of common EIP Information Elements.

4. Efficient mechanism for the processing of the EIP header (both in software and in hardware) can be defined when the different EIP Information Elements are carried inside the same EIP header.

4. Review of S~~s~~tandardized and ~~proposed ~~Proposed ~~evolutions ~~Evolutions of IPv6

In the last few years, we have witnessed important innovations in IPv6 networking, centered around the emergence of Segment Routing for IPv6 (SRv6) [RFC8754] and of the SRv6 "Network Programming model" [RFC8986]. With SRv6 it is possible to insert a _Network program_, i.e., a sequence of instructions (called _segments_), in a header of the IPv6 protocol, called Segment Routing Header (SRH).

Commenté [BMI14]: Already mentioned in Section 2

Another recent activity that proposed to extend the networking layer to support more complex functions, concerns the network monitoring. The concept of INT ("In-band Network Telemetry~~"~~) has been proposed since 2015 [onf-int] in the context of the definition of use cases for P4 based data plane programmability~~. The latest version of INT specifications dates November 2020~~ [int-spec]. [int-spec] specifies the format of headers that carry monitoring instructions and monitoring information along with data plane packets. The specific location for INT Headers is intentionally not specified: an INT Header can be inserted as an option or payload of any encapsulation type. The ~~In-band Telemetry~~INT concept has been adopted by the IPPM IETF Working Group, renaming it "In-situ Operations, Administration, and Maintenance" (IOAM)~~.~~ ~~The internet draft~~Indeed, [I-D.ietf-ippm-ioam-data]~~ is about to become an IETF RFC~~. Note that IOAM ~~is ~~focused on "limited domains" as defined in [RFC8799]. The in-situ OAM data fields can be encapsulated in a variety of protocols, including IPv6. The specification details for carrying IOAM data inside IPv6 headers are provided in draft [I-D.ietf-ippm-ioam-ipv6-options], which is also close to becoming an RFC. In particular, IOAM data fields can be encapsulated in IPv6 using either Hop-by-Hop Options header or Destination options header.

Another example of extensions to IPv6 for network monitoring is specified in [RFC8250], which defines an IPv6 Destination Options header called Performance and Diagnostic Metrics (PDM). The PDM option header provides sequence numbers and timing information as a basis for measurements.

The "Alternate Marking Method" is a recently proposed performance measurement approach described in [RFC8321]. The draft [I-D.draft-ietf-6man-ipv6-alt-mark] (also close to becoming an RFC) defines a new Hop-by-Hop Option to support this approach.

"Path Tracing" [I-D.draft-filsfils-spring-path-tracing] proposes an efficient solution for recording the route taken by a packet (including timestamps and load information taken at each hop along the route). This solution needs a new Hop-by-Hop Option to be defined.

[RFC8558] analyses the evolution of transport protocols. It recommends that explicit signals should be used when the endpoints desire that network elements along the path become aware of events related to ~~trasport~~transport protocol. Among the solutions, [RFC8558] considers the use of explicit signals at the network layer, and in particular it mentions that IPv6 hop-by-hop headers might suit this purpose.

~~The Internet Draft~~ [I-D.draft-ietf-6man-mtu-option] specifies a new IPv6 Hop-by-Hop option that is used to record the minimum Path MTU between a source and a destination. This draft is close to become an RFC.

~~The Internet Draft~~ [I-D.draft-ietf-6man-enhanced-vpn-vtn-id] proposes a new Hop-by-Hop option of IPv6 extension header to carry the Virtual Transport Network (VTN) identifier, which could be used to identify the set of network resources allocated to a VTN and the rules for packet processing. The procedure of processing the VTN option is also specified.

4.1. Considerations on Hop-by-hop Options allocation

We have listed several proposals or already standardized solutions that use the IPv6 Hop-by-Hop Options. These Options are represented with an ~~8~~ 8-bit~~s~~ code. The first two bits represent the action to be taken if the Options is unknown to a node that receives it, the third bit is used to specify if the content of the Options can be changed in flight. In particular, the Option Types that start with 001 should be ignored if unknown and can be changed in flight, which is the most common combination. The current IANA allocation for Option Types starting with 001 is (see https://www.iana.org/assignments/ipv6-parameters/ipv6-parameters.xhtml)

32 possible Option Types starting with 001
2 allocated by RFCs
2 temporary allocated by Internet Drafts
1 allocated for RFC3692-style Experiment
27 not allocated

We observe that there is a potential scarcity of the code points, as there are many scenarios that could require the definition of a new Hop-by-hop option. We also observe that having only 1 code point allocated for experiments is a very restrictive limitation.

~~5. 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.~~

6. Security Considerations

TODO Security

7. IANA Considerations

~~The definition of the EIP header as an Option for IPv6 Hop-by-hop Extension header requires the allocation of a codepoint from the "Destination Options and Hop-by-Hop Options" registry in the "Internet Protocol Version 6 (IPv6) Parameters" (https://www.iana.org/assignments/ipv6-parameters/ipv6-parameters.xhtm).~~

~~The definition of the EIP header as a TLV in the Segment Routing~~

~~Header requires the allocation of a codepoint from the "Segment Routing Header TLVs" registry in the "Internet Protocol Version 6 (IPv6) Parameters" (https://www.iana.org/assignments/ipv6-parameters/ipv6-parameters.xhtm).~~

~~The definition of EIP Information Elements in the EIP header will require the definition of a IANA registry.~~This document does not make any IANA request.

8. References

8.1. 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/info/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/info/rfc8174>.

8.2. Informative References

[I-D.draft-filsfils-spring-path-tracing]
Filsfils, C., Abdelsalam, A., Camarillo, P., Yufit, M., Graf, T., Su, Y., Matsushima, S., Valentine, M., and A. Dhamija, "Path Tracing in SRv6 networks", Work in Progress, Internet-Draft, draft-filsfils-spring-path-tracing-02, 16 August 2022, <https://www.ietf.org/archive/id/draft-filsfils-spring-path-tracing-02.txt>.

[I-D.draft-ietf-6man-enhanced-vpn-vtn-id]
Dong, J., Li, Z., Xie, C., Ma, C., and G. S. Mishra, "Carrying Virtual Transport Network (VTN) Information in IPv6 Extension Header", Work in Progress, Internet-Draft, draft-ietf-6man-enhanced-vpn-vtn-id-02, 24 October 2022, <https://www.ietf.org/archive/id/draft-ietf-6man-enhanced-vpn-vtn-id-02.txt>.

[I-D.draft-ietf-6man-ipv6-alt-mark]
Fioccola, G., Zhou, T., Cociglio, M., Qin, F., and R. Pang, "IPv6 Application of the Alternate Marking Method", Work in Progress, Internet-Draft, draft-ietf-6man-ipv6-alt-mark-17, 27 September 2022, <https://www.ietf.org/archive/id/draft-ietf-6man-ipv6-alt-mark-17.txt>.

[I-D.draft-ietf-6man-mtu-option]
Hinden, R. M. and G. Fairhurst, "IPv6 Minimum Path MTU Hop-by-Hop Option", Work in Progress, Internet-Draft, draft-ietf-6man-mtu-option-15, 10 May 2022, <https://www.ietf.org/archive/id/draft-ietf-6man-mtu-option-15.txt>.

[I-D.ietf-ippm-ioam-data]
Brockners, F., Bhandari, S., and T. Mizrahi, "Data Fields

for In Situ Operations, Administration, and Maintenance (IOAM)", Work in Progress, Internet-Draft, draft-ietf-ippm-ioam-data-17, 13 December 2021, <https://www.ietf.org/archive/id/draft-ietf-ippm-ioam-data-17.txt>.

[I-D.ietf-ippm-ioam-ipv6-options] Bhandari, S. and F. Brockners, "In-situ OAM IPv6 Options", Work in Progress, Internet-Draft, draft-ietf-ippm-ioam-ipv6-options-09, 11 October 2022, <https://www.ietf.org/archive/id/draft-ietf-ippm-ioam-ipv6-options-09.txt>.

[id-eip-headers] Salsano, S. and H. ElBakoury, "Extensible In-band Processing (EIP) Headers Definitions", 2022, <https://eip-home.github.io/eip-headers/draft-eip-headers-definitions.txt>.

[id-eip-use-cases] Salsano, S. and H. ElBakoury, "Extensible In-band Processing (EIP) Use Cases", 2022, <https://eip-home.github.io/use-cases/draft-eip-use-cases.txt>.

[int-spec] Group, T. P. A. W., "In-band Network Telemetry (INT) Dataplane Specification, version 2.1", n.d., <https://p4.org/p4-spec/docs/INT v2 1.pdf>.

[onf-int] P4.org, "Improving Network Monitoring and Management with Programmable Data Planes", 2015, <https://opennetworking.org/news-and-events/blog/improving-network-monitoring-and-management-with-programmable-data-planes/>.

[RFC8250] Elkins, N., Hamilton, R., and M. Ackermann, "IPv6 Performance and Diagnostic Metrics (PDM) Destination Option", RFC 8250, DOI 10.17487/RFC8250, September 2017, <https://www.rfc-editor.org/info/rfc8250>.

[RFC8321] Fioccola, G., Ed., Capello, A., Cociglio, M., Castaldelli, L., Chen, M., Zheng, L., Mirsky, G., and T. Mizrahi, "Alternate-Marking Method for Passive and Hybrid Performance Monitoring", RFC 8321, DOI 10.17487/RFC8321, January 2018, <https://www.rfc-editor.org/info/rfc8321>.

[RFC8558] Hardie, T., Ed., "Transport Protocol Path Signals", RFC 8558, DOI 10.17487/RFC8558, April 2019, <https://www.rfc-editor.org/info/rfc8558>.

[RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <https://www.rfc-editor.org/info/rfc8754>.

[RFC8799] Carpenter, B. and B. Liu, "Limited Domains and Internet Protocols", RFC 8799, DOI 10.17487/RFC8799, July 2020, <https://www.rfc-editor.org/info/rfc8799>.

[RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 10.17487/RFC8986, February 2021, <https://www.rfc-editor.org/info/rfc8986>.

Acknowledgments

TODO acknowledge.

Authors' Addresses

Stefano Salsano
Univ. of Rome Tor Vergata / CNIT
Email: stefano.salsano@uniroma2.it

Hesham ElBakoury
Consultant
Email: helbakoury@gmail.com

Diego R. Lopez
Telefonica, I+D
Email: diego.r.lopez@telefonica.com