

Internet Area Working Group  
Internet-Draft  
Intended status: Informational  
Expires: January 13, 2022

Y. Jia  
D. Trossen  
L. Iannone  
Huawei  
N. Shenoy  
R.I.T.  
P. Mendes  
Airbus  
July 12, 2021

**Gap Analysis in Internet Addressing**  
**draft-jia-intarea-internet-addressing-gap-analysis-00**

Abstract

There exist many extensions to Internet addressing, as it is defined in [RFC0791] for IPv4 and [RFC8200] for IPv6, respectively. Those extensions have been developed to fill gaps in capabilities beyond the basic properties of Internet addressing. This document outlines those properties as a baseline against which the extensions are categorized in terms of methodology used to fill the gap together with examples of solutions doing so.

While introducing such extensions, we outline the issues we see with those extensions. This ultimately leads to consider whether or not a more consistent approach to tackling the identified gaps, beyond point-wise extensions as done so far, would be beneficial. The benefits are the ones detailed in the companion document [I-D.jia-intarea-scenarios-problems-addressing], where, leveraging on the gaps identified in this memo and scenarios provided in [I-D.jia-intarea-scenarios-problems-addressing], a clear problem statement is provided.

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 January 13, 2022.

Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- [1. Introduction](#) . . . . . [3](#)
- [2. Properties of Internet Addressing](#) . . . . . [3](#)
  - [2.1. Property 1: Fixed Address Length](#) . . . . . [4](#)
  - [2.2. Property 2: Ambiguous Address Semantic](#) . . . . . [4](#)
  - [2.3. Property 3: Limited Address Semantic Support](#) . . . . . [4](#)
- [3. Filling Gaps through Extensions to Internet Addressing Properties](#) . . . . . [5](#)
  - [3.1. Length Extensions](#) . . . . . [5](#)
    - [3.1.1. Shorter Address Length](#) . . . . . [5](#)
    - [3.1.2. Longer Address Length](#) . . . . . [7](#)
    - [3.1.3. Summary](#) . . . . . [8](#)
  - [3.2. Identity Extensions](#) . . . . . [8](#)
    - [3.2.1. Anonymous Address Identity](#) . . . . . [9](#)
    - [3.2.2. Authenticated Address Identity](#) . . . . . [12](#)
    - [3.2.3. Summary](#) . . . . . [13](#)
  - [3.3. Semantic Extensions](#) . . . . . [14](#)
    - [3.3.1. Utilizing Extended Address Semantics](#) . . . . . [14](#)
    - [3.3.2. Utilizing Existing or Extended Header Semantics](#) . . . . . [17](#)
    - [3.3.3. Summary](#) . . . . . [20](#)
- [4. Overview of Approaches to Extend Internet Addressing](#) . . . . . [21](#)
- [5. Issues in Extensions to Internet Addressing](#) . . . . . [22](#)
  - [5.1. Limiting Address Semantics](#) . . . . . [22](#)
  - [5.2. Complexity and Efficiency](#) . . . . . [22](#)
    - [5.2.1. Repetitive encapsulation](#) . . . . . [23](#)
    - [5.2.2. Compounding issues with header compression](#) . . . . . [24](#)
    - [5.2.3. Introducing Path Stretch](#) . . . . . [24](#)
    - [5.2.4. Complicating Traffic Engineering](#) . . . . . [24](#)
  - [5.3. Security](#) . . . . . [25](#)
  - [5.4. Fragility](#) . . . . . [25](#)



- [6.](#) Summary of issues in Extensions to Internet Addressing . . . [26](#)
- [7.](#) Conclusions . . . . . [28](#)
- [8.](#) Security Considerations . . . . . [29](#)
- [9.](#) IANA Considerations . . . . . [30](#)
- [10.](#) Informative References . . . . . [30](#)
- Acknowledgments . . . . . [38](#)
- Authors' Addresses . . . . . [38](#)

**1. Introduction**

[I-D.jia-intarea-scenarios-problems-addressing] outlines scenarios and problems in Internet addressing through presenting a number of cases of communication that have emerged over the many years of utilizing the Internet and for which various extensions to the network interface-centric addressing of IPv6 have been developed. In order to continue the discussion on the emerging needs for addressing, initiated with [I-D.jia-intarea-scenarios-problems-addressing], this memo aims at identifying gaps between the Internet addressing model and desirable features that have been added by various extensions, in various contexts.

Our approach to identifying the gaps is guided by key properties of Internet addressing, outlined in [Section 2](#), namely (i) the fixed length of the IP addresses, (ii) the ambiguity of IP addresses semantic, while still (iii) providing limited IP address semantic support only. We derive those properties directly as a consequence of the respective standards that provide the basis for Internet addressing, most notably [[RFC0791](#)] for IPv4 and [[RFC8200](#)] for IPv6, respectively.

Those basic properties, and the potential issues that arise from those properties, give way to extensions that have been proposed over the course of deploying new Internet technologies. [Section 3](#) discusses those extensions, summarized as gaps against the basic properties in [Section 4](#).

Finally, we outline issues that arise with the extension-driven approach to the basic Internet addressing, discussed in [Section 5](#). We argue that any requirements for solutions that would revise the basic Internet addressing would require to address those issues.

**2. Properties of Internet Addressing**

As the Internet Protocol adoption has grown towards the global communication system we know today, its characteristics have evolved subtly, with [[RFC6250](#)] documenting various aspects of the IP service model and its frequent misconceptions, including Internet addressing.



In this section, we articulate the three most acknowledged properties related to `_Internet addressing_`. Those are (i) fixed IP address length, (ii) ambiguous IP address semantic, and (iii) limited IP address semantic support.

In [Section 3](#), we elaborate on various extensions that aim to expand Internet addressing beyond those properties; we position those extensions as intentions to close perceived gaps against those key properties.

### **[2.1.](#) Property 1: Fixed Address Length**

The fixed IP address length is specified as a key property of the design of Internet addressing, with 32 bits for IPv4 ([\[RFC0791\]](#)), and 128 bits for IPv6 ([\[RFC8200\]](#)), respectively. Given the capability of the hardware at the time of IPv4 design, a fixed length address was considered as a more appropriate choice for efficient packet forwarding. Although the address length was once considered to be variable during the design of Internet Protocol Next Generation ("IPng", cf., [\[RFC1752\]](#)) in 1990s, it finally inherited the design of IPv4 and adopted a fixed length address towards the current IPv6. As a consequence, the 128-bit fixed address length of IPv6 is regarded as a balance between fast forwarding (i.e., fixed length) and practically boundless cyberspace (128-bit length).

### **[2.2.](#) Property 2: Ambiguous Address Semantic**

Initially, the meaning of an IP address has been to identify an interface on a network device, although, when [\[RFC0791\]](#) was written, there were no explicit definitions of the IP address semantic.

With the global expansion of the Internet protocol, the semantic of the IP address is commonly believed to contain at least two notions, i.e., the explicit "locator", and the implicit "identifier". Because of the increasing use of IP addresses to both identify a node and to indicate the physical or virtual location of the node, the intertwined address semantics of identifier and locator was then gradually observed and first documented in [\[RFC2101\]](#) as "locator/identifier overload" property. With this, the IP address is used as an identification for host and server, very often directly used, e.g., for remote access or maintenance.

### **[2.3.](#) Property 3: Limited Address Semantic Support**

Although IPv4 [\[RFC0791\]](#) did not add any semantic to IP addresses beyond interface identification (and location), time has proven that additional semantics are desirable (c.f., the history of 127/8 [\[HISTORY127\]](#) or the introduction of private addresses [\[RFC1918\]](#)),



hence, IPv6 [[RFC4291](#)] introduced some form of additional semantics based on specific prefix values, for instance link-local addresses or a more structured multicast addressing. Nevertheless, systematic support for rich address semantics remains limited and basically prefix-based.

### **3. Filling Gaps through Extensions to Internet Addressing Properties**

Over the years, a plethora of extensions has been proposed in order to move beyond the native properties of IP addresses. We interpret the development of those extensions as filling gaps between the original properties of Internet addressing, outlined in the previous section, and desired new capabilities that those developing the extensions identified as being needed and desirable.

#### **3.1. Length Extensions**

Extensions in this subsection aim at extending the property described in [Section 2.1](#), i.e., fixed IP address length.

When IPv6 was designed, the main objective was to create an address space that would not lead to the same situation as IPv4, namely to address exhaustion. To this end, while keeping the same addressing model like IPv4, IPv6 adopted a 128-bit address length with the aim of providing a sufficient and future-proof address space. The choice was also founded on the assumption that advances in hardware and Moore's law would still allow to make routing and forwarding faster, and IPv6 routing table manageable.

We observe, however, that the rise of new use cases but also the number of new, e.g., industrial/home or small footprint devices, was possibly unforeseen. Sensor networks and more generally the Internet of Things (IoT) emerged after the core body of work in IPv6. Nevertheless, it was clear from the beginning that differently from IPv6 assumptions, 128-bit addresses were costly in certain scenarios.

At the same time, new use cases emerged, where addresses even bigger than 128-bit are useful, like in the context of content centric networks, structured addresses, or cryptographically generated addresses.

##### **3.1.1. Shorter Address Length**

###### **3.1.1.1. Description:**

In the context of IoT [[RFC7228](#)], where bandwidth and energy are very scarce resources, the static length of 128-bit for an IP address is more a hindrance than a benefit since 128-bit for an IP address may





occupy a lot of space, even to the point of being the dominant part of a packet. In order to use bandwidth more efficiently and use less energy in end-to-end communication, solutions have been proposed that allow for very small network layer headers instead.

#### **3.1.1.2. Methodology:**

**\*Header Compression/Translation\*:** One of the main approaches to reduce header size in the IoT context is by header compression. Such technique is based on a stateful approach, utilizing what is usually called a "context" on the IoT sensor and the gateway for communications between an IoT device and a server placed somewhere in the Internet - from the edge to the cloud.

The role of the "context" is to provide a way to "compress" the original IP header in a smaller one, by using shorter address information and/or dropping some field, using the context as a kind of dictionary.

**\*Separate device from locator identifier\*:** Approaches that can offer customized address length that is adequate for use in such constrained domains are preferred. Using different namespaces for the 'device identifier' and the 'routing' or 'locator identifier' is one such approach.

#### **3.1.1.3. Examples**

**\*Header Compression/Translation\*:** Considering one base station is supposed to serve hundreds of user devices, maximizing the effectiveness for specific spectrum directly improve user quality of experience. To achieve the optimal utilization of the spectrum resource in wireless area, the RObust Header Compression (ROHC) [[RFC5795](#)] mechanism, which has been widely adopted in cellular network like WCDMA, LTE, and 5G, utilizes header compression to shrink existing IPv6 headers onto shorter ones.

Similarly, header compression techniques for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN) have been around for several years now, constituting a main example of using the notion of a shared context in order to reduce the size of the network layer header ([[RFC6282](#)], [[RFC7400](#)], [[ITU9959](#)]). More recently, other compression solution have been proposed for Low Power Wide Area Networks (LPWAN - [[RFC8376](#)]), but they basically rely on the same "context" approach ([[RFC8724](#)]).

Also, constraints coming from either devices or carrier links would lead to mixed scenarios and compound requirements for extraordinary header compression. For native IPv6 communications on DECT ULE and



MS/TP Networks [[RFC6282](#)], dedicated compression mechanisms are specified in [[RFC8105](#)] and [[RFC8163](#)], while the transmission of IPv6 packets over NFC and PLC, specifications are being developed in [[I-D.ietf-6lo-nfc](#)] and [[I-D.ietf-6lo-plc](#)].

\*Separate device from locator identifier\*: Solutions such as proposed in [[EIBP](#)] and [[I-D.ietf-lisp-rfc6830bis](#)] can utilize a separation of device from locator, where only the latter is used for routing between the different domains using the same technology, therefore enabling the use of shorter addresses in the (possibly constrained) local environment. Device IDs used within such domains are carried as part of the payload by EIBP and hence can be of shorter size suited to the domain, while, for instance, in LISP a flexible address encoding [[RFC8060](#)] allows shorter addresses to be supported in the LISP control plane [[I-D.ietf-lisp-rfc6833bis](#)].

### **[3.1.2.](#) Longer Address Length**

#### **[3.1.2.1.](#) Description**

To address current networking demands it may be preferred to use variable and longer address lengths. For example in content delivery networks, longer addresses such as URLs are required to fetch content, an approach that information-centric networking (ICN) applied for any data packet sent in the network, using information-based addressing. Further, as an approach to address the routing challenges faced in the Internet, structured addresses may be used, avoiding the need for routing protocols.

Further, as an approach to address the routing challenges such as scalability, stability and the complex interworking between OSPF, eBGP and iBGP faced in the Internet, structured addresses may be used, avoiding the need for routing protocols overall.

#### **[3.1.2.2.](#) Methodology**

The crucial need for longer address lengths comes from `_semantic extensions_` to IP addresses, where the extensions themselves do not fit within the length limitation of the IP address. [Section 3.3](#) discusses those extensions, divided into those where limitations stemming from the IP address lengths are considered and those where information is either represented through a longer address and/or additional header information.



**3.1.2.3. Examples**

See [Section 3.3](#) for examples of solutions that consider extended address and header information to represent extended semantics that are not limited by the IP address length.

**3.1.3. Summary**

Table 1, summarize the methodologies and the examples towards filling the gaps on length extensions.

	Methodology	Examples
Shorter Address Length	Header compression/translation	6LOWPAN, ROHC
	Separate device from locator identifier	EIBP, LISP, ILNP, HIP
Longer Address Length	Same as Semantic Extension Table 3	/

Table 1: Summary Length Extensions

**3.2. Identity Extensions**

Extensions in this subsection try extending the property described in [Section 2.2](#), i.e., "locator/identifier overload" of the ambiguous address semantic.

From the perspective of Internet users, on the one hand, the implicit identifier semantic result in the privacy issue of network behavior tracking and association. Despite that IP address assignment may be dynamic, they are nowadays considered as a personal data and as such undergoes privacy protection regulations like General Data Protection Regulation ("GDPR", cf. [\[GDPR\]](#)). Hence, additional mechanism are necessary in order to protect end user privacy.

For network regulation of sensitive information, on the other hand, dynamically allocated IP addresses are not sufficient to guarantee device or user identification. As such, different address allocation systems, with stronger identification properties are necessary where security and authentication are at highest priority. Hence, in order to protect information security within a network, additional mechanism are necessary to identify the users or the devices attached to the network.



### **3.2.1. Anonymous Address Identity**

#### **3.2.1.1. Description**

As discussed in [Section 2.2](#), IP addresses reveal both 'network locations' as well as implicit 'identifier' information to both traversed network elements and destination nodes alike. This enables recording, correlation, and profiling of user behaviors and historical network traces, possibly down to individual real user identity. The IETF, e.g., in [[RFC7258](#)], has taken a clear stand on preventing any such pervasive monitoring means by classifying them as an attack on end users' right to be left along (i.e., privacy). Regulations such as the EU's General Data Protection Regulation (GDPR) classifies, for instance, the 'online identifier' as personal data which must be carefully protected; this includes end users' IP addresses [[GDPR](#)].

Even before pervasive monitoring [[RFC7258](#)], IP addresses have been seen as something that some organizational owners of networked system may not want to reveal at the individual level towards any non-member of the same organization.

#### **3.2.1.2. Methodology:**

\*Traffic Proxy\*: Detouring the traffic to a trusted proxy is a heuristic solution. Since nodes between trusted proxy and destination (including the destination per se) can only observe the source address of the proxy, the "identification" of the origin source can thereby be hidden. To confuse the nodes between origin and the proxy, the traffic on such route would be encrypted via a key negotiated either in-band or off-band. Considering that all applications' traffic in such route can be seen as a unique flow directed to the same "unknown" node, i.e., the trusted proxy, eavesdroppers in such route have to make more efforts to correlate user behavior through statistical analysis even if they are capable of identifying the users via their source addresses. The protection lays in the inability to isolate single application specific flows. According to the methodology, such approach is IP version independent and works for both IPv4 and IPv6.

\*Source Address Rollover\*: Privacy issues related to address "identifier" semantic can be mitigated through regular change (beyond the typical 24 hours lease of DHCP). Built on the inevitabilities of "identifier" that IP address carries, such approach promote to change the source IP address at certain frequency. Under such methodology, refresh cycling window may reach to a balance between privacy protection and address update cost. Due to the limited space that IPv4 contains, such approach usually works for IPv6 only.





**\*Private Address Spaces\*:** Their introduction in [[RFC1918](#)] foresaw private addresses (assigned to specific address spaces by the IANA) as a means to communicate purely locally, e.g., within an enterprise, by separating private from public IP addresses. Considering that private addresses are never directly reachable from the Internet, hosts adopting private addresses are invisible and thus "anonymous" for the Internet. Besides, hosts for purely local communication used the latter while hosts requiring public Internet service access would still use public IP addresses.

**\*Address Translation\*:** The aforementioned original intention for using private IP addresses, namely for purely local communication, resulted in a lack of flexibility in changing from local to public Internet access on the basis of what application would require which type of service.

If eventually every end-system in an organization would require some form of public Internet access in addition to local one, an adequate number of public Internet addresses would be required for providing to all end systems. Instead, address translation enables to utilize many private IP addresses within an organization, while only relying on one (or few) public IP addresses for the overall organization.

In principle, address translation can be applied recursively. This can be seen in modern broadband access where Internet providers may rely on carrier-grade address translation for all their broadband customers, who in turn employ address translation of their internal home or office addresses to those (private again) IP addresses assigned to them by their network provider.

Two benefits arise from the use of (private to public IP) address translation, namely (i) the hiding of local end systems at the level of the (address) assigned organization, and (ii) the reduction of public IP addresses necessary for communication across the Internet. While the latter has been seen for long as a driver for address translation, we focus on the first issue in this section, also since we see such privacy benefit as well as objective as still being valid in addressing systems like IPv6 where address scarcity is all but gone [[GNATCATCHER](#)].

**\*Separate device from locator identifier\*:** Solutions that make a clear separation between the routing locator and the identifier, can allow for a device ID of any size, which in turn can be encrypted by a network element deployed at the border of routing domain (e.g., a router). Both source and end domain addresses can be encrypted and transported, as in the routing domain, only the routing locator is used.



### **3.2.1.3. Examples:**

**\*Traffic Proxy\*:** Although not initially designed for traffic proxy, Virtual Private Network (VPN, cf. [[VPN](#)]) is widely utilized for packets origin hiding as a traffic detouring methodology. As it evolved, VPN derivatives like WireGuard [[WireGuard](#)] become a mainstream instance for user privacy and security enhancement.

With such methodology in mind, onion routing [[ONION](#)], instantiated in the TOR Project [[TOR](#)], achieves high anonymity through traffic hand over via intermediates before reaching the destination. Since the architecture of TOR requires at least three proxies in traffic, none of them is aware of the entire route. Given that the proxies themselves can be deployed all over the cyberspace, trust will not be the prerequisite if proxies are randomly selected.

In addition, dedicated protocols are also expected to be customized for privacy improvement via traffic proxy. For example, Oblivious DNS over HTTPS (ODOH, cf. [[ODOH](#)]) use a third-part proxy to obscure identifications of user source addresses during DNS over HTTPS (DoH, cf. [[RFC8484](#)]) resolution. Similarly, Oblivious HTTP [[OHTTP](#)] involve proxy alike in the HTTP environment.

**\*Source Address Rollover\*:** As for source address rollover, it has been standardized that IP addresses for Internet users should be dynamic and temporary every time they are being generated [[RFC8981](#)]. This benefits from the available address space in the case of IPv6, through which address generation or assignment should be unpredictable and stochastic for outside observers.

More radically, [[EPHEMERALv6](#)] advocates an 'ephemeral address' for each process that change over time. Through this, correlating user behaviors conducted by different identifiers (i.e., source address) becomes hard, if not impossible, based on the IP packet header alone.

**\*Private Addresses\*:** The use and assignment of private address for IPv4 is laid out in [[RFC1918](#)], while unique local addresses (ULAs) in IPv6 [[RFC4193](#)] take over the role of private address spaces in IPv4.

**\*Network Address Translation\*:** The translation from one IP address realm into another is laid out in [[RFC2663](#)], using a stateful address binding to translate between the realms, e.g., from a private into a public address space. As outlined in [[RFC2663](#)], basic address translation is usually extended to include port information in the translation process, support bidirectional or simple outbound traffic only. Given address translation can be performed several times in cascade, NATs may exist as part of existing customer premise equipment (CPE), such as a cable or an Ethernet router, with private



wired/wireless connectivity, or may be provided in a carrier environment to further translate ISP-internal private addresses to a pool of (assigned) public IP addresses. The latter is often dynamically assigned to CPEs upon bootstrapping of the CPE.

*\*Separate device from locator identifier\**: EIBP [[EIBP](#)] utilizes a structured approach to addressing. It separates the routing ID from the device ID, where only the routing ID is used for routing. Hence the device IDs can be encrypted. Similarly, LISP uses separate namespaces for routing and identification allowing to "hide" identifiers in encrypted LISP packets that expose only known routing information [[RFC8061](#)].

### **[3.2.2. Authenticated Address Identity](#)**

#### **[3.2.2.1. Description](#)**

In some scenarios (e.g., corporate networks) it desirable to being able authenticate IP addresses so to prevent malicious attackers to spoof IP addresses. This is usually achieved by using a mechanism that allows to prove ownership of the IP address.

#### **[3.2.2.2. Methodology](#)**

*\*Self-certified addresses\**: This method is usually based on the use of nodes' public/private keys. A node creates its own interface ID (IID) by using a cryptographic hash of its public key (with some additional parameters). Messages are then signed using the nodes' private key. The destination of the message will verify the signature through the information in the IP address. Self-certification has the advantage that no third party or additional security infrastructure is needed. Any node can generate its own address locally and then only the address and the public key are needed to verify the binding between the public key and the address.

*\*Third party granted addresses\**: DHCP (Dynamic Host Configuration Protocol) is widely used to provide IP addresses, however, in its basic form, it does not perform any check and even an unauthorized user without the right to use the network can obtain an IP address. To solve this problem, a trusted third party has to grant access to the network before generating an address (via DHCP or other) that identifies the user. User authentication done securely either based on physical parameters like MAC addresses or based on an explicit login/password mechanism.



**3.2.2.3. Examples**

\*Self-certified Addresses\*: Major example of this methodology is [RFC3972], where IPv6 cryptographically Generated Addresses (CGA) are defined. The original specs have been already amended (cf., [RFC4581] and [RFC4982]) in order to support multiple (stronger) cryptographic algorithms.

\*Third party granted addresses\*: [RFC3118] defines a DHCP option through which authorization tickets can be generated and newly attached hosts with proper authorization can be automatically configured from an authenticated DHCP server. Solutions exists where separate servers are used for user authentication ([UA-DHCP], [RFC4014])

**3.2.3. Summary**

Table 2, summarize the methodologies and the examples towards filling the gaps on identity extensions.

	Methodology	Examples
Anonymous Address Identity	Traffic Proxy	VPN, TOR, ODoH
	Source Address Rollover	SLAAC
	Private Address Spaces	ULA
	Address Translation	NAT
	Separate device from locator identifier	EIBP, LISP
	Authenticated Address Identity	Self-certified Addresses
Third party granted addresses		DHCP-Option

Table 2: Summary Identity Extensions





### **3.3. Semantic Extensions**

Extensions in this subsection try extending the property described in [Section 2.3](#), i.e., limited address semantic support.

As explained in [Section 2.2](#), IP addresses carry both locator and interface identification semantic. Some efforts exist in trying to separate these semantics either in different address spaces or through different address format. Beyond just identification, location, and the fixed address size, other efforts extended the semantic through existing or additional header fields (or header options) outside the Internet address.

#### **3.3.1. Utilizing Extended Address Semantics**

##### **3.3.1.1. Description**

Several extensions have been developed to extend beyond the limited IPv6 semantics. Those approaches may include to apply structure to the address, utilize specific prefixes or entirely utilize the IPv6 address for different semantics while re-encapsulating the original packet to restore the semantics in another part of the network. For instance, structured addresses have the capability to introduce delimiters to identify semantic information in the header, therefore not constraining any semantic by size limitations of the address fields. We note here that extensions often start out as being proposed as an extended header semantic, while standardization may drive the solution to adopt an approach to accommodate their semantic within the limitations of an IP address. This section does include examples of this kind.

##### **3.3.1.2. Methodology**

*\*Semantic prefixes\**: Semantic prefixes are used to separate the IPv6 address space. Through this, new address families, such as for information-centric networking [[HICN](#)], service routing or other semantically rich addressing, can be defined, albeit limited by the prefix length and structure as well as the overall length limitation of the IPv6 address.

*\*Separate device from locator identifier\**: The option to use separate namespaces for the device address (w.r.t. the router locator), would offer more freedom for the use of different semantics.

*\*Structured addressing\**: One approach to address the routing challenges faced in the internet proposes the use of structured addresses, to void the need for routing protocols. Benefits of this approach can be significant, with the structured addresses capturing



the relative physical or virtual position of routers in the network, and being variable in length. Key to the approach, however, is that the structured addresses capturing the relative physical or virtual position of routers in the network, or networks in an internetwork may not fit within the fixed and limited IP address length (cf., [Section 3.1.2](#)). Other approaches to structured approaches may be the definition of application-specific structured binary components as a generalization of the URL schema used for HTTP-level communication.

*\*Localized forwarding semantics\**: Layer 2 hardware, such as SDN switches, are limited to the use of specific header fields for forwarding decisions. Hence, devising new localized forwarding mechanisms may make use of existing header fields such as the IPv6 source/destination field to achieve the desired forwarding behavior, while encapsulating the original IPv6 packets into the payload in order to be restored at the local forwarding network boundary. Network sizes in those solutions are limited by the size of the utilized address field, e.g., 256 bits for IPv6, thereby limiting the size of networks where such techniques could be used. The IPv6 address information provided in the encapsulated packet purely serves identifying the network endpoint, not its location.

#### **3.3.1.3. Examples**

*\*Semantic prefixes\**: Newer approaches to IP anycast suggest the use of service identification in combination with a binding IP address model [[SFCANYCAST](#)] as a way to allow for metric-based traffic steering decisions; approaches for Service Function Chaining (SFC) [[RFC7665](#)] utilize the next service header (NSH) information and packet classification to determine the destination of the next packet hop.

Another example of the usage of different packet header extensions based in IP addressing is Segment Routing. In this case the source chooses a path and encodes it in the packet header as an ordered list of segments. Segments are coded at header of IPv6 Packet, based on the definition of a new Routing Extensions Header type, the Segment Routing Header (SRH), which contains Segment List. This is similar to what is already specified in [RFC2460](#). Packets are then transmitted from source node with a list of segment ID (SID) inserted at header List dictates the path to follow in the network. Such segment ID are coded as 128 bit IPv6 addresses.

Approaches such as [[HICN](#)] utilize semantic prefixing to allow for ICN forwarding behavior within an IPv6 network. In this case an HICN name is the hierarchical concatenation of a name prefix and a name suffix, in which the name prefix is encoded as an IPv6 128 bits word and carried in IPv6 header fields, while the name suffix is encoded



in transport headers fields such as TCP. However, it is a challenge to determine which IPv6 prefixes should be used as name prefixes. In order to know which IPv6 packets should be interpreted based on an ICN semantic, it is desirable to be able to recognize that an IPv6 prefix is a name prefix, e.g. to define a specific address family (AF\_HICN, b0001::/16). This establishment of a specific address family allows the management and control plane to locally configure HICN prefixes and announce them to neighbors for interconnection.

\*Separate device from locator identifier\*: EIBP [[EIBP](#)] separates the routing locator from the device identifier, relaxing therefore any semantic constraints on the device identifier. Similarly, LISP uses a flexible encoding named LISP Canonical Address Format (LCAF [[RFC8061](#)]), which allows to associate to routing locators any possible form (and length) of identifier. ILNP [[RFC6740](#)] introduces as well a different semantic of IP addresses, while aligning to the IPv6 address format (128 bits). Basically, ILNP introduces a sharper logical separation between the 64 most significant bits and the 64 least significant bits of an IPv6 address. The former being a global locator, while the latter being an identifier that can have different semantic (rather than just being an interface identifier).

\*Structured addressing\*: Network topology captures the physical connectivity among devices in the network. There is a structure associated with the topology. Examples are the core-distribution-access router structure commonly used in enterprise networks and clos topologies that are used to provide multiple connections between Top of Rack (ToR) devices and multiple layers of spine devices. Internet service providers use a tier structure that defines their business relationships. A clear structure of connected networks can be noticed in the internet. EIBP [[EIBP](#)] proposes to leverage the physical structure (or a virtual structure overlaid on the physical structure) to auto assign addresses to routers in a network or networks in an internetwork to capture their relative position in the physical/virtual topology. EIBP proposes to administratively identify routers/networks with a tier value based on the structure.

\*Localized forwarding semantics\*: Approaches such as those outlined in [[REED](#)] suggest using a novel forwarding semantic based on path information carried in the packet itself, said path information consists in a fixed size bitfield (see [[REED](#)] for more information on how to represent the path information in said bitfield). In order to utilize existing, e.g., SDN-based, forwarding switches, the direct use of the IPv6 source/destination address is suggested for building appropriate match-action rules (over the suitable binary information representing the local output ports), while preserving the original IPv6 information in the encapsulated packet. As mentioned above, such use of the existing IPv6 address fields limits the size of the



network to a maximum of 256 bits (therefore paths in the network over which such packets can be forwarded). [ICNIP], however, goes a step further by suggesting to use the local forwarding as direct network layer mechanism, removing the IP packet and only leaving the transport/application layer, with the path identifier constituting the network-level identifier albeit limited by using the existing IP header for backward compatibility reason (the next section outlines the removal of this limitation).

### **3.3.2. Utilizing Existing or Extended Header Semantics**

#### **3.3.2.1. Description:**

While the former sub-section explored extended address semantic, thereby limiting any such extended semantic with that of the existing IPv6 semantic and length, additional semantics may also be placed into the header of the packet or the packet itself, utilized for the forwarding decision to the appropriate endpoint according to the extended semantic.

Reasons for embedding such new semantics may be related to traffic engineering since it has long been shown that the IP address itself is not enough to steer traffic properly since the IP address itself is not semantically rich enough to adequately describe the forwarding decision to be taken in the network, not only impacting WHERE the packet will need to go but also HOW it will need to be sent.

#### **3.3.2.2. Methodology:**

**\*In-Header extensions\*:** One way to add additional semantic besides the address fields is to use other fields already present in the header.

**\*Headers option extensions\*:** Another mechanism to add additional semantic is to actually add additional fields, e.g., through Header Options in IPv4 or through Extension Headers in IPv6.

**\*Re-encapsulation extension\*:** A more radical approach for additional semantic is the use of a complete new header that is designed so to carry the desired semantic in an efficient way (often as a shim header).

**\*Structured addressing\*:** Similar to the methodology that structures addresses within the limitations of the IPv6 address length, outlined in the previous sub-sections, structured addressing can also be applied within existing or extended header semantics, e.g., utilizing a dedicated (extension) header to carry the structured address information.





**\*Localized forwarding semantics\*:** This set of solutions apply capabilities of newer (programmable) forwarding technology, such as [P4], to utilize any header information for a localized forwarding decision. This removes any limitation to use existing header or address information for embedding a new address semantic into the transferred packet.

### **3.3.2.3. Examples:**

**\*In-Header extensions\*:** In order to allow additional semantic with respect to the pure Internet addressing, the original design of IPv4 included the field "Type of Service" [RFC2474], while IPv6 introduced the "Flow label" and the "Traffic Class" [RFC8200]. In a certain way, those fields can be considered "semantic extensions" of IP addresses, and they are "in-header" because natively present in the IP header (differently from options and extension headers). However, they proved not to be sufficient. Very often a variety of network operation are performed on the well-known 5-tuple (source and destination addresses; source and destination port number; and protocol number). In some contexts all of the above mentioned fields are used in order to have a very fine grained solution ([RFC8939]).

**\*Headers option extensions\*:** Header options have been largely under exploited in IPv4. However, the introduction of the more efficient extension header model in IPv6 along with technology progress made the use of header extensions more widespread in IPv6. Segment Routing re-introduced the possibility to add path semantic to the packet by encoding a loosely defined source routing ([RFC8402]). Similarly, in the aim to overcome the inherent shortcoming of the multi-homing in the IP context, SHIM6 ([RFC5533]) also proposed the use of an extension header able to carry multi-homing information which cannot be accommodated natively in the IPv6 header.

To serve a moving endpoint, mechanisms like Mobile IPv6 [RFC6275] are used for the maintenance of connection continuity by a dedicated IPv6 extension header. In such a case, the IP address of the home agent in Mobile IPv6 is basically an identification of the on-going communication. In order to go beyond the interface identification model of IP, the Host Identity Protocol (HIP) tries to introduce an identification layer to provide (as the name says) host identification. The whole architecture relies on the use of another type of extension header [RFC7401].

**\*Re-encapsulation extension\*:** Differently from the previous approach, re-encapsulation prepends complete new IP headers to the original packet introducing a completely custom shim header between the outer and inner header. This is the case for LISP, adding a LISP specific header right after an IP+UDP header ([I-D.ietf-lisp-rfc6830bis]). A



similar design is used by VXLAN ([RFC7348]) and GENEVE ([RFC8926]), even if they are designed for a data center context. IP packets can also be wrapped with headers using more generic and semantically rich names, for instance with ICN [ICNIP].

*\*Structured addressing:* Solutions such as those described in the previous sub-section, e.g., EIBP [EIBP], can provide structured addresses that are not limited to the IPv6 address length but instead carry the information in an extension header to remove such limitation.

Also information-centric networking (ICN) naming approaches usually introduce structures in the (information) names without limiting themselves to the IP address length; more so, ICN proposes its own header format and therefore radically breaks with not only IP addressing semantic but the format of the packet header overall. For this, approaches such as those described in [RFC8609] define a TLV-based binary application component structure that is carried as a 'name' part of the CCN messages. Such a name is a hierarchical structure for identifying and locating a data object, which contains a sequence of name components. Names are coded based on 2-level nested Type-Length-Value (TLV) encodings, where the name-type field in the outer TLV indicates this is a name, while the inner TLVs are name components including a generic name component, an implicit SHA-256 digest component and a SHA-256 digest of Interest parameters. For textual representation, URIs are normally used to represent names, as defined in [RFC3986].

In geographic addressing, position based routing protocols use the geographic location of nodes as their addresses, and packets are forwarded when possible in a greedy manner towards the destination. For this purpose the packet header includes a field coding the geographic coordinates (x, y, z) of the destination node, as defined in [RFC2009]. Some proposals also rely on extra fields in the packet header to code the distance towards the destination, in which case only the geographic coordinates of neighbors are exchanged. This way the location of the destination is protected even if routing packets are eavesdropped.

*\*Localized forwarding semantics:* Unlike the original suggestion in [REED] to use existing SDN switches, the proliferation of P4 [P4] opens up the possibility to utilize a locally limited address semantic, expressed through the path identifier, as an entirely new header (including its new address) with an encapsulation of the IP packet for E2E delivery (including further delivery outside the localized forwarding network or positioning the limited address semantic directly as the network address semantic for the packet, i.e., removing any IP packet encapsulation from the forwarded packet,



as done in [ICNIP]. Removing the IPv6 address size limitation by not utilizing the existing IP header for the forwarding decision also allows for extensible length approaches for building the path identifier with the potential for increasing the supported network size. On the downside, this approach requires to encapsulate the original IP packet header for communication beyond the local domain in which the new header is being used, such as discussed in the previous point above on 're-encapsulation extension'.

**3.3.3. Summary**

Table 3, summarize the methodologies and the examples towards filling the gaps on semantic extensions.

	Methodology	Examples
Utilizing Extended Address Semantics	Semantic prefixes	HICN
	Separate device from locator identifier	EIBP, ILNP, LISP, HIP
	Structured addressing	EIBP, ILNP
	Localized forwarding semantics	REED
Utilizing Existing or Extended Header Semantics	In-Header extensions	DetNet
	Headers option extensions	SHIM6, SRv6, HIP
	Re-encapsulation extension	VxLAN, ICNIP
	Structured addressing	EIBP
	Localized forwarding semantics	REED

Table 3: Summary Semantic Extensions



**4. Overview of Approaches to Extend Internet Addressing**

From the view of the extensions, Table 4 describes the objectives these extensions in extending the properties of Internet addressing. As summarized, extensions may aim to extend one property of the Internet addressing, or extend other properties at the same time.

	Length Extension	Identity Extension	Semantic Extension
6LoWPAN	x		
ROHC	x		
TOR		x	
ODoH		x	
SLAAC		x	
CGA		x	x
NAT	x	x	
HICN		x	x
ICNIP	x	x	x
CCNx names	x	x	x
EIBP	x	x	x
Geo addressing	x		x
REED	x (with P4)		x
DetNet		x	
Mobile IP			x
SHIM6			x
SRV6			x
HIP		x	x





VxLAN		x	x	
LISP		x	x	
SFC		x	x	
+-----+-----+-----+-----+-----+				

Table 4: Relationship between Extensions and Internet Addressing

## 5. Issues in Extensions to Internet Addressing

While the extensions to the original Internet properties discussed in [Section 3](#) demonstrate the flexibility of the basic proposed mechanisms, they also bring with them a number of issues, which we discuss in the following section. For this, we outline the problems caused, linking those to the approaches to extensions summarized in [Section 4](#).

### 5.1. Limiting Address Semantics

Many approaches changing the semantics of communication, e.g., through separating host identification from network node identification [[RFC7401](#)], separating the device identifier from the routing locator ([[EIBP](#)], [[I-D.ietf-lisp-introduction](#)]), or through identifying content and services directly [[HICN](#)], are limited by the existing packet size and semantic constraints of IPv6, e.g., in the form of its source and destination network addresses.

While approaches such as [[ICNIP](#)] may override the addressing semantics, e.g., by replacing IPv6 source and destination information with path identification, a possible unawareness of endpoints still requires the carrying of other address information as part of the payload.

Also, the expressible service or content semantic may be limited, as in [[HICN](#)] or the size of supported networks [[REED](#)] due to relying on the limited bit positions usable in IPv6 addresses.

### 5.2. Complexity and Efficiency

A crucial issue is the additional complexity introduced for realizing the additional addressing semantics. This is particularly an issue since we see those additional semantics particularly at the edge of the Internet, utilizing the existing addressing semantic of the Internet to interconnect the domains that require those additional semantics.



Furthermore, any additional complexity can often come with an efficiency and cost penalty, particularly at the edge of the network, where resource constraints may play a significant role. Compression processes, taking ROHC as an example, require additional resources both for the sender generating the compressed header but also the gateway linking to the general Internet by re-establishing the full IP header (cf., [[ROHC](#)]).

Conversely, the performance requirements of core networks, in terms of packet processing speed, make the accommodation of extensions to addressing often prohibitive. This is not only due to the necessary extra processing that is specific to the extension, but also due to the complexity that will need to be managed in doing so at significantly higher speeds than at the edge of the network. The observations on the dropping of packets with IPv6 extension headers in the real world is (partially) due to such a implementation complexity [[RFC7872](#)].

Another example for lowering the efficiency of packet forwarding is the routing in systems like TOR [[TOR](#)]. As detailed before, traffic in TOR, for anonymity purposes, should be handed over by at least three intermediates before reaching the destination. Thus, frequent relaying enhances the privacy, however, because such kind of solutions have to be implemented at application level, which comes at the cost of lower communication efficiency. May be a different privacy enhanced address semantic would enable efficient implementation of TOR-like solutions at network layer.

### **5.2.1. Repetitive encapsulation**

Repetitive encapsulation is an issue 'bloating' packet sizes with additional encapsulation headers. Addressing proposals such as those in [[ICNIP](#)] utilize path identification within an alternative forwarding architecture that acts upon the provided path identification. However, due to the limitation of existing flow-based architectures with respect to the supported header structures (in the form of IPv4 or IPv6 headers), the new routing semantics are being inserted into the existing header structure, while repeating the original, sender-generated header structure, in the payload of the packet as it traverses the local domain, effectively doubling the header overhead per packet.

The problem is also present in a number of solutions tackling different issues, e.g., mobility [[I-D.ietf-lisp-mn](#)], DC networking ([[RFC8926](#)], [[RFC7348](#)], [[I-D.ietf-intarea-gue](#)]), traffic engineering [[RFC8986](#)], and privacy ([[TOR](#)], [[SPHINX](#)]). Certainly these solutions are able to avoid other issues, like path lengthening or privacy issues, as described later, but they come at the price of multiple



encapsulations that reduce the effective payload. This not only hampers efficiency in terms of header-to-payload ratio but also introduces "encapsulation points", which in turn add complexity to the (often edge) network as well as fragility due to introducing possible failure points; we discuss this aspect more in [Section 5.4](#).

### **[5.2.2](#). Compounding issues with header compression**

IP header overhead requires header compression in constrained environments, such as wireless sensor networks and IoT in general. Together with fragmentation, both tasks constitute significant energy consumption, as shown in [[HEADER\\_COMP\\_ISSUES1](#)], negatively impacting reduced function devices that often rely on battery for operation. Further, the reliance on the compression/decompression points creates a dependence on such gateways, which may be a problem for intermittent scenarios.

According to implementation of `_contiki-ng_` [[CONTIKI](#)], an example of operating system for IoT devices, the source codes for 6LowPan requires at least 600kb to include a header compression process. In certain use cases, such requirement can be an obstacle for extremely constrained devices, especially for the RAM and energy consumption.

### **[5.2.3](#). Introducing Path Stretch**

Mobile IP [[RFC6275](#)], which was designed for connection continuity in the face of moving endpoints, is a typical case for path stretch. Since traffic must follow a triangular route before arriving at the destination, such detour routing inevitably impacts transmission efficiency as well as latency.

### **[5.2.4](#). Complicating Traffic Engineering**

While many extensions to the original IP address semantic target to enrich the decisions that can be taken to steer traffic according to requirements for, e.g., QoS, mobility, chaining, compute/network metrics, flow treatment, path usage, etc., the realization of the mechanisms as individual solutions likely complicates the original goal of traffic engineering when individual solutions are being used in combination. Ultimately, this may even prevent the combined use of more than one mechanism and/or policies with a need to identify and prevent incompatibilities of mechanisms. Key here is not the issue arising from using conflicting traffic engineering policies but conflicting realizations of policies that may well generally work well alongside ([[ROBUSTSDN](#)], [[TRANSACTIONSDN](#)]).

This not only increases fragility, as we will discuss separately in [Section 5.4](#), but also requires careful planning for which mechanisms



to use and in which combination, likely needing human-in-the-loop approaches alongside possible automation approaches for the individual solutions.

### 5.3. Security

The properties described in [Section 2](#) have, obviously, also consequences in terms of security and privacy related issues, as already mentioned in other parts of this document.

For instance, in the effort of being somehow backward compatible, HIP [[RFC7401](#)] uses 128-bit Host Identity, which may be not sufficiently cryptographically strong in the future because of the limited size (future computational power may erode 128-bit security). Similarly, CGA [[RFC3972](#)] also align to the 128-bit limit but may use only 59 bits of them, hence, the packet signature may not be sufficiently robust to attacks [[I-D.rafiee-6man-cga-attack](#)].

IP addresses, even temporary ones meant to protect privacy, have been long recognized as a "Personal Identification Information" that allows even to geolocate the communicating endpoints [[RFC8280](#)]. The use of temporary addresses provides sufficient privacy protection only if the renewal rate is high [[EPHEMERALv6](#)]. However, this causes additional issues, like the large overhead due to the Duplicate Address Detection, the impact on the Neighbor Discovery mechanism, in particular the cache, which can even lead to communication disruption. With such drawbacks, the extensions may even lead to defeat the target actually lowering security rather than increasing it.

The introduction of alternative addressing semantics has also been used to help in (D)DoS attacks mitigation. This leverages on changing the service identification model so to avoid topological information exposure, making the potential disruptions likely remain limited [[ADDRLESS](#)]. However, this increased robustness to DDoS comes at the price of important communication setup latency and fragility, as discussed next.

### 5.4. Fragility

From the extensions discussed in [Section 3](#), it is evident that having alternative or additional address semantic and formats available for making routing as well as forwarding decisions dependent on these, is common place in the Internet. This, however, adds many extension-specific translation/adaptation points, mapping the semantic and format in one context into what is meaningful in another context, but also, more importantly, creating a dependency towards an additional





component, often without explicit exposure to the endpoints that originally intended to communicate.

For instance, the re-writing of IP addresses to facilitate the use of private address spaces throughout the public Internet, realized through network address translators (NATs), conflicts with the end-to-end nature of communication between two endpoints. Additional (flow) state is required at the NAT middlebox to smoothly allow communication, which in turn creates a dependency between the NAT and the end-to-end communication between those endpoints, thus increasing the fragility of the communication relation.

A similar situation arises when supporting constrained environments through a header compression mechanism, adding the need for, e.g., a ROHC [RFC5795] element in the communication path, with communication-related compression state being held outside the communicating endpoints. Failure will introduce some inefficiencies due to context regeneration, which may affects the communicating endpoints, increasing fragility of the system overall.

Such translation/adaptation between semantic extensions to the original 'semantic' of an IP address is generally not avoidable when accommodating more than a single universal semantic. However, the solution-specific nature of every single extension is likely to noticeably increase the fragility of the overall system, since individual extensions will need to interact with other extensions that may be deployed in parallel, but were not designed taking into account such deployment scenario (cf., [I-D.ietf-intarea-tunnels]). Considering that extensions to traditional per-hop-behavior (based on IP addresses) can essentially be realized over almost 'any' packet field, the possible number of conflicting behaviors or diverging interpretation of the semantic and/or content of such fields, among different extensions, may soon become an issue, requiring careful testing and delineation at the boundaries of the network within which the specific extension has been realized.

**6. Summary of issues in Extensions to Internet Addressing**

Table 5, derived from Section 5, summarize the issues related to each of the extension. As summarized, each extension involves at least one issue. For extension like ICNIP, several issues may be involved at the same time. Further, as explained in Section 5.4, "fragility" does not mean that the extensions are fragile per-se, they are robust for the task they have been designed for, rather we refer to the existence of additional components that may fail (as Murphy's law teach us), hence, making the overall system more fragile.

+-----+-----+-----+-----+-----+



	Limiting Address Semantics	Complexity and Efficiency	Security	Fragility
6LoWPAN		x		x
ROHC		x		x
TOR		x		x
ODoH		x		
SLAAC		x		
CGA	x		x	
NAT		x		x
HICN	x			
ICNIP	x	x		
CCNx name	x			
EIBP	x	x		
Geo addressing	x			x
REED	x			
DetNet		x		
Mobile IP		x		x
SHIM6				x
SRv6				x
HIP			x	x
VxLAN		x		
LISP	x	x		
SFC		x		x



Table 5: Issues in Extensions to Internet Addressing

## 7. Conclusions

The examples of extensions discussed in [Section 3](#) to the original Internet addressing scheme show that extensibility beyond the original model (and its underlying per-hop behavior) is a desired capability for networking technologies and has been so for a long time. Generally, we can observe that those extensions are driven by the requirements of stakeholders, expecting a desirable extended functionality from the introduction of the specific extension. If interoperability is required, those extensions require standardization of possibly new fields, new semantics as well as (network and/or end system) operations alike.

One could argue that the power of the packet header (and the entire packet for that matter) lies in its potential to be used for extensions that satisfy the various requirements of those developing the extensions with given deployments and use cases in mind. The author in [\[EXPRESSIVE-POWER\]](#) argues that this capability represents the 'expressive power of the Internet design', with developments such as Software-Defined Networking (SDN) enabling per-hop operations (to be specified in dedicated match-action rules) that take more than just the IP address as input, even beyond the strict IP header, albeit still limited to certain packet header(s) fields, as in the OpenFlow specification [\[OPENFLOW\]](#).

Beyond the fragility, discussed in this document, caused by adding, overriding, amending semantics and behaviors (and deploying them in a running system), such approach to extensibility that opens up other header fields may, however, also be limiting the realization of per-hop operations. As an example, although the authors in [\[REED\]](#) utilize the power of SDN to enable extension-specific per-hop behaviors that follow path-based rather than endpoint locator semantics, they do so within the limitation of currently supported header fields in SDN. The result is the overriding of the IPv6 source/destination address with the semantic input (here the path forwarding identifier) for the desired per-hop behavior, which in turn creates the need for border translation to avoid 'leaking' such (malformed) IPv6-like packet into a standard IPv6 network. Leaving the IPv6 address information intact albeit unused is not possible due to the lack of accommodating the extended per-hop behavior information needed for the proposed solution.

Since the publication of [\[REED\]](#) (and its realization in existing real world trials), the newer P4 technology has removed this limitation, therefore positioning the entire packet content as input into per-hop



behavior and removing the need for overriding existing packet fields and restoring their semantics at the borders of the network.

We see such evolution of per-hop behavior implementation as closing the loop to approaches to addressing considered in early days of the packet-based networking [[RFC0757](#)][POSTEL], based on structured and flexible addresses, similar to the heap or stack model suggestions found in [[EXPRESSIVE-POWER](#)]. We believe that approaching addressing from such desire to enable extensions by design rather than through a pure extension engineering approach would also address possible fragility and security issues due to unwanted interactions between possibly incompatible extensions, while also enabling more efficient realization of the extensions through a common execution point in the forwarding element.

The issues we identified with the extension-specific solution approach, point to the need for a discussion on Internet addressing, as formulated in the companion document [[I-D.jia-intarea-scenarios-problems-addressing](#)] that formalizes the problem statement through scenarios that highlight the shortcomings of the Internet addressing model.

It is our conclusion that the existence of the many extensions to the original Internet addressing is clear evidence for gaps that have been identified over time by the wider Internet community: it is time to develop an architectural but more importantly a sustainable approach to make Internet addressing extensible in order to capture the many new uses we will still identify for the Internet to come. Any inaction on our side in that regard will only compound the issues we identified, eventually hampering the future Internet's readiness for those new uses.

## **8. Security Considerations**

The present memo does not introduce any new technology and/or mechanism and as such does not introduce any security threat to the TCP/IP protocol suite.

As an additional note, and as we discussed in this document, security and privacy aspects were not considered as part of the key properties for Internet addressing, which led to the introduction of a number of extensions intending to fix those gaps. The analysis presented in this memo (non-exhaustively) shows those issues are either solved in an ad-hoc manner at application level, or at transport layer, while at network level only few extensions tackling specific aspects exist albeit often with limitations due to the adherence to the Internet addressing model and its properties.





## 9. IANA Considerations

This document does not include any IANA request.

## 10. Informative References

### [ADDRLESS]

Hao, S., Liu, R., Weng, Z., Chang, D., Bao, C., and X. Li, "Addressless: A new internet server model to prevent network scanning", PLOS ONE Vol. 16, pp. e0246293, DOI 10.1371/journal.pone.0246293, February 2021.

### [CONTIKI]

"Contiki-NG: The OS for Next Generation IoT Devices", n.d., <<https://github.com/contiki-ng/contiki-ng>>.

### [EIBP]

Willis, N., "A Structured Approach to Routing in the Internet", June 2021, <First Intl Workshop on Semantic Addressing and Routing for Future Networks>.

### [EPHEMERALv6]

Gont, F. and G. Gont, "IPv6 Addressing Considerations", [draft-gont-v6ops-ipv6-addressing-considerations-01](#) (work in progress), February 2021.

### [EXPRESSIVE-POWER]

Clark, D., "The expressive power of the Internet design", April 2009, <<http://groups.csail.mit.edu/ana/People/DDC/Expressive%20power%205.pdf>>.

### [GDPR]

Voigt, P. and A. von dem Bussche, "The EU General Data Protection Regulation (GDPR)", Springer International Publishing book, DOI 10.1007/978-3-319-57959-7, 2017.

### [GNATCATCHER]

"Global Network Address Translation Combined with Audited and Trusted CDN or HTTP-Proxy Eliminating Reidentification", n.d., <<https://github.com/bslassey/ip-blindness>>.

### [HEADER\_COMP\_ISSUES1]

Mesrinejad, F., Hashim, F., Noordin, N., Rasid, M., and R. Abdullah, "The effect of fragmentation and header compression on IP-based sensor networks (6LOWPAN)", The 17th Asia Pacific Conference on Communications, DOI 10.1109/apcc.2011.6152926, October 2011.



- [HICN] Muscariello, L., "Hybrid Information-Centric Networking: ICN inside the Internet Protocol", March 2018, <<https://datatracker.ietf.org/meeting/interim-2018-icnrg-01/materials/slides-interim-2018-icnrg-01-sessa-hybrid-icn-hicn-luca-muscariello>>.
- [HISTORY127] "History of 127/8 as localhost/loopback addresses", n.d., <<https://elists.isoc.org/pipermail/internet-history/2021-January/006920.html>>.
- [I-D.ietf-6lo-nfc] Choi, Y., Hong, Y., Youn, J., Kim, D., and J. Choi, "Transmission of IPv6 Packets over Near Field Communication", [draft-ietf-6lo-nfc-17](#) (work in progress), August 2020.
- [I-D.ietf-6lo-plc] Hou, J., Liu, B., Hong, Y., Tang, X., and C. E. Perkins, "Transmission of IPv6 Packets over PLC Networks", [draft-ietf-6lo-plc-06](#) (work in progress), April 2021.
- [I-D.ietf-intarea-gue] Herbert, T., Yong, L., and O. Zia, "Generic UDP Encapsulation", [draft-ietf-intarea-gue-09](#) (work in progress), October 2019.
- [I-D.ietf-intarea-tunnels] Touch, J. and M. Townsley, "IP Tunnels in the Internet Architecture", [draft-ietf-intarea-tunnels-10](#) (work in progress), September 2019.
- [I-D.ietf-lisp-introduction] Cabellos, A. and D. Saucez, "An Architectural Introduction to the Locator/ID Separation Protocol (LISP)", [draft-ietf-lisp-introduction-13](#) (work in progress), April 2015.
- [I-D.ietf-lisp-mn] Farinacci, D., Lewis, D., Meyer, D., and C. White, "LISP Mobile Node", [draft-ietf-lisp-mn-09](#) (work in progress), February 2021.
- [I-D.ietf-lisp-rfc6830bis] Farinacci, D., Fuller, V., Meyer, D., Lewis, D., and A. Cabellos, "The Locator/ID Separation Protocol (LISP)", [draft-ietf-lisp-rfc6830bis-36](#) (work in progress), November 2020.



## [I-D.ietf-lisp-rfc6833bis]

Farinacci, D., Maino, F., Fuller, V., and A. Cabellos, "Locator/ID Separation Protocol (LISP) Control-Plane", [draft-ietf-lisp-rfc6833bis-30](#) (work in progress), November 2020.

## [I-D.jia-intarea-scenarios-problems-addressing]

Jia, Y., Trossen, D., Iannone, L., 3rd, D. E. E., and P. Liu, "Challenging Scenarios and Problems in Internet Addressing", [draft-jia-intarea-scenarios-problems-addressing-00](#) (work in progress), February 2021.

## [I-D.rafiiee-6man-cga-attack]

Rafiiee, H. and C. Meinel, "Possible Attack on Cryptographically Generated Addresses (CGA)", [draft-rafiiee-6man-cga-attack-03](#) (work in progress), May 2015.

## [ICNIP]

Trossen, D., Robitzsch, S., Reed, M., Al-Naday, M., and J. Riihijarvi, "Internet Services over ICN in 5G LAN Environments", [draft-trossen-icnrg-internet-icn-5gln-04](#) (work in progress), October 2020.

## [ITU9959]

Badenhop, C., Fuller, J., Hall, J., Ramsey, B., and M. Rice, "Evaluating ITU-T G.9959 Based Wireless Systems Used in Critical Infrastructure Assets", IFIP Advances in Information and Communication Technology pp. 209-227, DOI 10.1007/978-3-319-26567-4\_13, 2015.

## [ODoH]

Kinnear, E., McManus, P., Pauly, T., Verma, T., and C. A. Wood, "Oblivious DNS Over HTTPS", [draft-pauly-dprive-oblivious-doh-06](#) (work in progress), March 2021.

## [OHTTP]

Thomson, M. and C. A. Wood, "Oblivious HTTP", [draft-thomson-http-oblivious-01](#) (work in progress), February 2021.

## [ONION]

Goldschlag, D., Reed, M., and P. Syverson, "Onion routing", Communications of the ACM Vol. 42, pp. 39-41, DOI 10.1145/293411.293443, February 1999.

## [OPENFLOW]

McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., and J. Turner, "OpenFlow: enabling innovation in campus networks", ACM SIGCOMM Computer Communication Review Vol. 38, pp. 69-74, DOI 10.1145/1355734.1355746, March 2008.



- [P4] Bosshart, P., Daly, D., Gibb, G., Izzard, M., McKeown, N., Rexford, J., Schlesinger, C., Talayco, D., Vahdat, A., Varghese, G., and D. Walker, "P4: programming protocol-independent packet processors", ACM SIGCOMM Computer Communication Review Vol. 44, pp. 87-95, DOI 10.1145/2656877.2656890, July 2014.
- [POSTEL] Postel, J., "Internetwork Protocol Approaches", IEEE Transactions on Communications Vol. 28, pp. 604-611, DOI 10.1109/tcom.1980.1094705, April 1980.
- [REED] Reed, M., Al-Naday, M., Thomos, N., Trossen, D., Petropoulos, G., and S. Spirou, "Stateless multicast switching in software defined networks", 2016 IEEE International Conference on Communications (ICC), DOI 10.1109/icc.2016.7511036, May 2016.
- [RFC0757] Deutsch, D., "Suggested solution to the naming, addressing, and delivery problem for ARPANET message systems", [RFC 757](#), DOI 10.17487/RFC0757, September 1979, <<https://www.rfc-editor.org/info/rfc757>>.
- [RFC0791] Postel, J., "Internet Protocol", STD 5, [RFC 791](#), DOI 10.17487/RFC0791, September 1981, <<https://www.rfc-editor.org/info/rfc791>>.
- [RFC1752] Bradner, S. and A. Mankin, "The Recommendation for the IP Next Generation Protocol", [RFC 1752](#), DOI 10.17487/RFC1752, January 1995, <<https://www.rfc-editor.org/info/rfc1752>>.
- [RFC1918] Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G., and E. Lear, "Address Allocation for Private Internets", [BCP 5](#), [RFC 1918](#), DOI 10.17487/RFC1918, February 1996, <<https://www.rfc-editor.org/info/rfc1918>>.
- [RFC2009] Imielinski, T. and J. Navas, "GPS-Based Addressing and Routing", [RFC 2009](#), DOI 10.17487/RFC2009, November 1996, <<https://www.rfc-editor.org/info/rfc2009>>.
- [RFC2101] Carpenter, B., Crowcroft, J., and Y. Rekhter, "IPv4 Address Behaviour Today", [RFC 2101](#), DOI 10.17487/RFC2101, February 1997, <<https://www.rfc-editor.org/info/rfc2101>>.
- [RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", [RFC 2474](#), DOI 10.17487/RFC2474, December 1998, <<https://www.rfc-editor.org/info/rfc2474>>.





- [RFC2663] Srisuresh, P. and M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations", [RFC 2663](#), DOI 10.17487/RFC2663, August 1999, <<https://www.rfc-editor.org/info/rfc2663>>.
- [RFC3118] Droms, R., Ed. and W. Arbaugh, Ed., "Authentication for DHCP Messages", [RFC 3118](#), DOI 10.17487/RFC3118, June 2001, <<https://www.rfc-editor.org/info/rfc3118>>.
- [RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)", [RFC 3972](#), DOI 10.17487/RFC3972, March 2005, <<https://www.rfc-editor.org/info/rfc3972>>.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, [RFC 3986](#), DOI 10.17487/RFC3986, January 2005, <<https://www.rfc-editor.org/info/rfc3986>>.
- [RFC4014] Droms, R. and J. Schnizlein, "Remote Authentication Dial-In User Service (RADIUS) Attributes Suboption for the Dynamic Host Configuration Protocol (DHCP) Relay Agent Information Option", [RFC 4014](#), DOI 10.17487/RFC4014, February 2005, <<https://www.rfc-editor.org/info/rfc4014>>.
- [RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", [RFC 4193](#), DOI 10.17487/RFC4193, October 2005, <<https://www.rfc-editor.org/info/rfc4193>>.
- [RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", [RFC 4291](#), DOI 10.17487/RFC4291, February 2006, <<https://www.rfc-editor.org/info/rfc4291>>.
- [RFC4581] Bagnulo, M. and J. Arkko, "Cryptographically Generated Addresses (CGA) Extension Field Format", [RFC 4581](#), DOI 10.17487/RFC4581, October 2006, <<https://www.rfc-editor.org/info/rfc4581>>.
- [RFC4982] Bagnulo, M. and J. Arkko, "Support for Multiple Hash Algorithms in Cryptographically Generated Addresses (CGAs)", [RFC 4982](#), DOI 10.17487/RFC4982, July 2007, <<https://www.rfc-editor.org/info/rfc4982>>.
- [RFC5533] Nordmark, E. and M. Bagnulo, "Shim6: Level 3 Multihoming Shim Protocol for IPv6", [RFC 5533](#), DOI 10.17487/RFC5533, June 2009, <<https://www.rfc-editor.org/info/rfc5533>>.



- [RFC5795] Sandlund, K., Pelletier, G., and L-E. Jonsson, "The RObusT Header Compression (ROHC) Framework", [RFC 5795](#), DOI 10.17487/RFC5795, March 2010, <<https://www.rfc-editor.org/info/rfc5795>>.
- [RFC6250] Thaler, D., "Evolution of the IP Model", [RFC 6250](#), DOI 10.17487/RFC6250, May 2011, <<https://www.rfc-editor.org/info/rfc6250>>.
- [RFC6275] Perkins, C., Ed., Johnson, D., and J. Arkko, "Mobility Support in IPv6", [RFC 6275](#), DOI 10.17487/RFC6275, July 2011, <<https://www.rfc-editor.org/info/rfc6275>>.
- [RFC6282] Hui, J., Ed. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", [RFC 6282](#), DOI 10.17487/RFC6282, September 2011, <<https://www.rfc-editor.org/info/rfc6282>>.
- [RFC6740] Atkinson, R.J. and S.N. Bhatti, "Identifier-Locator Network Protocol (ILNP) Architectural Description", [RFC 6740](#), DOI 10.17487/RFC6740, November 2012, <<https://www.rfc-editor.org/info/rfc6740>>.
- [RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for Constrained-Node Networks", [RFC 7228](#), DOI 10.17487/RFC7228, May 2014, <<https://www.rfc-editor.org/info/rfc7228>>.
- [RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", [BCP 188](#), [RFC 7258](#), DOI 10.17487/RFC7258, May 2014, <<https://www.rfc-editor.org/info/rfc7258>>.
- [RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", [RFC 7348](#), DOI 10.17487/RFC7348, August 2014, <<https://www.rfc-editor.org/info/rfc7348>>.
- [RFC7400] Bormann, C., "6LoWPAN-GHC: Generic Header Compression for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", [RFC 7400](#), DOI 10.17487/RFC7400, November 2014, <<https://www.rfc-editor.org/info/rfc7400>>.
- [RFC7401] Moskowitz, R., Ed., Heer, T., Jokela, P., and T. Henderson, "Host Identity Protocol Version 2 (HIPv2)", [RFC 7401](#), DOI 10.17487/RFC7401, April 2015, <<https://www.rfc-editor.org/info/rfc7401>>.



- [RFC7665] Halpern, J., Ed. and C. Pignataro, Ed., "Service Function Chaining (SFC) Architecture", [RFC 7665](#), DOI 10.17487/RFC7665, October 2015, <<https://www.rfc-editor.org/info/rfc7665>>.
- [RFC7872] Gont, F., Linkova, J., Chown, T., and W. Liu, "Observations on the Dropping of Packets with IPv6 Extension Headers in the Real World", [RFC 7872](#), DOI 10.17487/RFC7872, June 2016, <<https://www.rfc-editor.org/info/rfc7872>>.
- [RFC8060] Farinacci, D., Meyer, D., and J. Snijders, "LISP Canonical Address Format (LCAF)", [RFC 8060](#), DOI 10.17487/RFC8060, February 2017, <<https://www.rfc-editor.org/info/rfc8060>>.
- [RFC8061] Farinacci, D. and B. Weis, "Locator/ID Separation Protocol (LISP) Data-Plane Confidentiality", [RFC 8061](#), DOI 10.17487/RFC8061, February 2017, <<https://www.rfc-editor.org/info/rfc8061>>.
- [RFC8105] Mariager, P., Petersen, J., Ed., Shelby, Z., Van de Logt, M., and D. Barthel, "Transmission of IPv6 Packets over Digital Enhanced Cordless Telecommunications (DECT) Ultra Low Energy (ULE)", [RFC 8105](#), DOI 10.17487/RFC8105, May 2017, <<https://www.rfc-editor.org/info/rfc8105>>.
- [RFC8163] Lynn, K., Ed., Martocci, J., Neilson, C., and S. Donaldson, "Transmission of IPv6 over Master-Slave/Token-Passing (MS/TP) Networks", [RFC 8163](#), DOI 10.17487/RFC8163, May 2017, <<https://www.rfc-editor.org/info/rfc8163>>.
- [RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, [RFC 8200](#), DOI 10.17487/RFC8200, July 2017, <<https://www.rfc-editor.org/info/rfc8200>>.
- [RFC8280] ten Oever, N. and C. Cath, "Research into Human Rights Protocol Considerations", [RFC 8280](#), DOI 10.17487/RFC8280, October 2017, <<https://www.rfc-editor.org/info/rfc8280>>.
- [RFC8376] Farrell, S., Ed., "Low-Power Wide Area Network (LPWAN) Overview", [RFC 8376](#), DOI 10.17487/RFC8376, May 2018, <<https://www.rfc-editor.org/info/rfc8376>>.
- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", [RFC 8402](#), DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.



- [RFC8484] Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", [RFC 8484](#), DOI 10.17487/RFC8484, October 2018, <<https://www.rfc-editor.org/info/rfc8484>>.
- [RFC8609] Mosko, M., Solis, I., and C. Wood, "Content-Centric Networking (CCNx) Messages in TLV Format", [RFC 8609](#), DOI 10.17487/RFC8609, July 2019, <<https://www.rfc-editor.org/info/rfc8609>>.
- [RFC8724] Minaburo, A., Toutain, L., Gomez, C., Barthel, D., and JC. Zuniga, "SCHC: Generic Framework for Static Context Header Compression and Fragmentation", [RFC 8724](#), DOI 10.17487/RFC8724, April 2020, <<https://www.rfc-editor.org/info/rfc8724>>.
- [RFC8926] Gross, J., Ed., Ganga, I., Ed., and T. Sridhar, Ed., "Geneve: Generic Network Virtualization Encapsulation", [RFC 8926](#), DOI 10.17487/RFC8926, November 2020, <<https://www.rfc-editor.org/info/rfc8926>>.
- [RFC8939] Varga, B., Ed., Farkas, J., Berger, L., Fedyk, D., and S. Bryant, "Deterministic Networking (DetNet) Data Plane: IP", [RFC 8939](#), DOI 10.17487/RFC8939, November 2020, <<https://www.rfc-editor.org/info/rfc8939>>.
- [RFC8981] Gont, F., Krishnan, S., Narten, T., and R. Draves, "Temporary Address Extensions for Stateless Address Autoconfiguration in IPv6", [RFC 8981](#), DOI 10.17487/RFC8981, February 2021, <<https://www.rfc-editor.org/info/rfc8981>>.
- [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>>.
- [ROBUSTSDN] Canini, M., Kuznetsov, P., Levin, D., and S. Schmid, "A distributed and robust SDN control plane for transactional network updates", 2015 IEEE Conference on Computer Communications (INFOCOM), DOI 10.1109/infocom.2015.7218382, April 2015.





- [ROHC] Fitzek, F., Rein, S., Seeling, P., and M. Reisslein, "RObust Header Compression (ROHC) Performance for Multimedia Transmission over 3G/4G Wireless Networks", Wireless Personal Communications Vol. 32, pp. 23-41, DOI 10.1007/s11277-005-7733-2, January 2005.
- [SFCANYCAST] Wion, A., Bouet, M., Iannone, L., and V. Conan, "Distributed Function Chaining with Anycast Routing", Proceedings of the 2019 ACM Symposium on SDN Research, DOI 10.1145/3314148.3314355, April 2019.
- [SPHINX] Danezis, G. and I. Goldberg, "Sphinx: A Compact and Provably Secure Mix Format", 2009 30th IEEE Symposium on Security and Privacy, DOI 10.1109/sp.2009.15, May 2009.
- [TOR] "The Tor Project", n.d., <<https://www.torproject.org/>>.
- [TRANSACTIONSNDN] Curic, M., Despotovic, Z., Hecker, A., and G. Carle, "Transactional Network Updates in SDN", 2018 European Conference on Networks and Communications (EuCNC), DOI 10.1109/eucnc.2018.8442793, June 2018.
- [UA-DHCP] Komori, T. and T. Saito, "The secure DHCP system with user authentication", 27th Annual IEEE Conference on Local Computer Networks, 2002. Proceedings. LCN 2002., DOI 10.1109/lcn.2002.1181774, n.d..
- [VPN] Khanvilkar, S. and A. Khokhar, "Virtual private networks: an overview with performance evaluation", IEEE Communications Magazine Vol. 42, pp. 146-154, DOI 10.1109/mcom.2004.1341273, October 2004.
- [WireGuard] Donenfeld, J., "WireGuard: Next Generation Kernel Network Tunnel", Proceedings 2017 Network and Distributed System Security Symposium, DOI 10.14722/ndss.2017.23160, 2017.

## Acknowledgments

Thanks to Carsten Bormann for useful conversations.

## Authors' Addresses



Yihao Jia  
Huawei Technologies Co., Ltd  
156 Beiqing Rd.  
Beijing 100095  
P.R. China

Email: [jiayihao@huawei.com](mailto:jiayihao@huawei.com)

Dirk Trossen  
Huawei Technologies Duesseldorf GmbH  
Riesstr. 25C  
Munich 80992  
Germany

Email: [dirk.trossen@huawei.com](mailto:dirk.trossen@huawei.com)

Luigi Iannone  
Huawei Technologies France S.A.S.U.  
18, Quai du Point du Jour  
Boulogne-Billancourt 92100  
France

Email: [luigi.iannone@huawei.com](mailto:luigi.iannone@huawei.com)

Nirmala Shenoy  
Rochester Institute of Technology  
New-York 14623  
USA

Email: [nxsvks@rit.edu](mailto:nxsvks@rit.edu)

Paulo Mendes  
Airbus  
Willy-Messerschmitt Strasse 1  
Munich 81663  
Germany

Email: [paulo.mendes@airbus.com](mailto:paulo.mendes@airbus.com)

