

Workgroup: Network Working Group
Internet-Draft:
draft-jia-flex-ip-address-structure-00
Published: 31 October 2020
Intended Status: Standards Track
Expires: 4 May 2021
Authors: Y. Jia Z. Chen S. Jiang
 Huawei Huawei Huawei

Flexible IP: An Adaptable IP Address Structure

Abstract

Along as the popularization and adoption of IP in emerging scenarios, challenges emerge as well due to the ossified address structure. To enable TCP/IP in networks that previously using exclusive protocol, a flexible address structure would be far more preferred for their particular properties [[draft-jia-scenarios-flexible-address-structure](#)].

This document describes a flexible address structure -- Flexible IP (FlexIP) acting on limited domains [[RFC8799](#)]. FlexIP is expected to proactively adapt scenarios under flexible address structure. Meanwhile, FlexIP still benefit from global reachability based on the IPv6 interoperability.

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 4 May 2021.

Copyright Notice

Copyright (c) 2020 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](#)
- [2. Terminology](#)
- [3. Targeted Scenario](#)
- [4. Design Considerations](#)
 - [4.1. Multi-Semantics](#)
 - [4.2. Elastic Address Space](#)
 - [4.3. Scalability](#)
 - [4.4. Interoperability](#)
- [5. FlexIP Address structure](#)
 - [5.1. Restrained Space Format](#)
 - [5.2. Extendable Space Format](#)
 - [5.3. Hierarchical Segments Format](#)
 - [5.4. Multi-Semantics Format](#)
- [6. FlexIP Address Text Representation](#)
- [7. Interoperability](#)
- [8. Security Considerations](#)
- [9. IANA Considerations](#)
- [10. Informative References](#)
- [Authors' Addresses](#)

1. Introduction

As Internet Protocol (IP) gradually turned into the "waist" of the "TCP/IP" protocol stack, it is considered to be the core pillar of the entire Internet [[waist](#)]. Although numerous technologies in this "TCP/IP" protocol stack have emerged, evolved, or obsoleted by others, the IPv6 technology [[RFC8200](#)] is the only forward in network layer along with the Internet upgrades. IPv6, as the unique successor of IPv4 [[RFC0791](#)] defined by IETF, fixes defects for most of its parts. Most notably, the address space is enormously expanded from 32-bit to 128-bit in IPv6 reformation. Despite that IPv6 is expected to serve almost infinite devices in the foreseeable future, several scenarios are found in trouble when IPv6 is in use.

For instance, due to the market and cost requirements, numerous Internet-of-things (IoTs) are devised to be tiny and resource constrained. However, such rigorous requirement induce manufactures to adopt lightweight protocols like bluetooth, other than TCP/IP, for inter communications [[iot](#)]. Energy consumption, which is sound

in most terminal cases, becomes the greatest challenge when introducing IPv6 to IoTs. Document [[draft-jia-scenarios-flexible-address-structure](#)] details the challenges for more cases of IPv6.

To conquer these challenges, several improvements are promoted by the corporation of related standard organizations. Document [[RFC4944](#)] depict the transmission of IPv6 over IEEE 802.15.4 network, and such a method enables indirectly support of IPv6 in IoTs, e.g., Thread Protocol [[thread](#)]. Besides, document [[RFC7668](#)] describe the fusion of IPv6 and Bluetooth Low Energy, and such a method also enable the communications among nodes with Bluetooth and IPv6. Although none of these proposal is superior on the basis of market sharing, all endeavour orientate to the comprehensive adoption of TCP/IP protocol stack in new communication cases.

This document proposes an adaptive IP address format -- Flexible IP (abbr. FlexIP) orienting emerging scenarios [[draft-jia-scenarios-flexible-address-structure](#)] within limited domains [[RFC8799](#)], and maintain global reachability with IPv6. In general, FlexIP is composed through a hierarchical, self-explanatory address structure. Compared to the patch-like solutions (e.g., [[RFC4944](#)], [[RFC8724](#)]) that passively tune IP to be compatible with different scenarios, FlexIP proactively makes address structure flexible enough to adapt to various network cases. This variation, opposite to the fixed form of IPv4/IPv6 address, is expected to make Internet Protocol agilely cover futuristic and unknown scenarios.

//TODO: more citations needed.

2. Terminology

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.

3. Targeted Scenario

As a architectural solution for scenarios detailed in [[draft-jia-scenarios-flexible-address-structure](#)], FlexIP is expected to be adopted in limited domains [[RFC8799](#)]. According to the definition in [[RFC8799](#)], limited domain refers to a single physical network attached to or running in parallel with the Internet, or a defined set of users and nodes distributed over a much wider area, but drawn together by a single virtual network over the Internet. Within the limited domains, requirements, behaviors, and semantics could be noticeable local and network specific.

[Figure 1](#) depicts the orientation of FlexIP in IPv6 and Internet. Generally, Internet with its backbone is principally composed by IPv6, and networks with IPv4 are recognized as legacy and attached at the edge of the Internet with transition mechanisms. The position of networks with FlexIP structure is quite similar as IPv4. For networks within limited domains, FlexIP can be marginally adopted at the edge of the Internet. Such network use FlexIP to fulfill proprietary capabilities, while retain global reachability through IPv6 via transition.

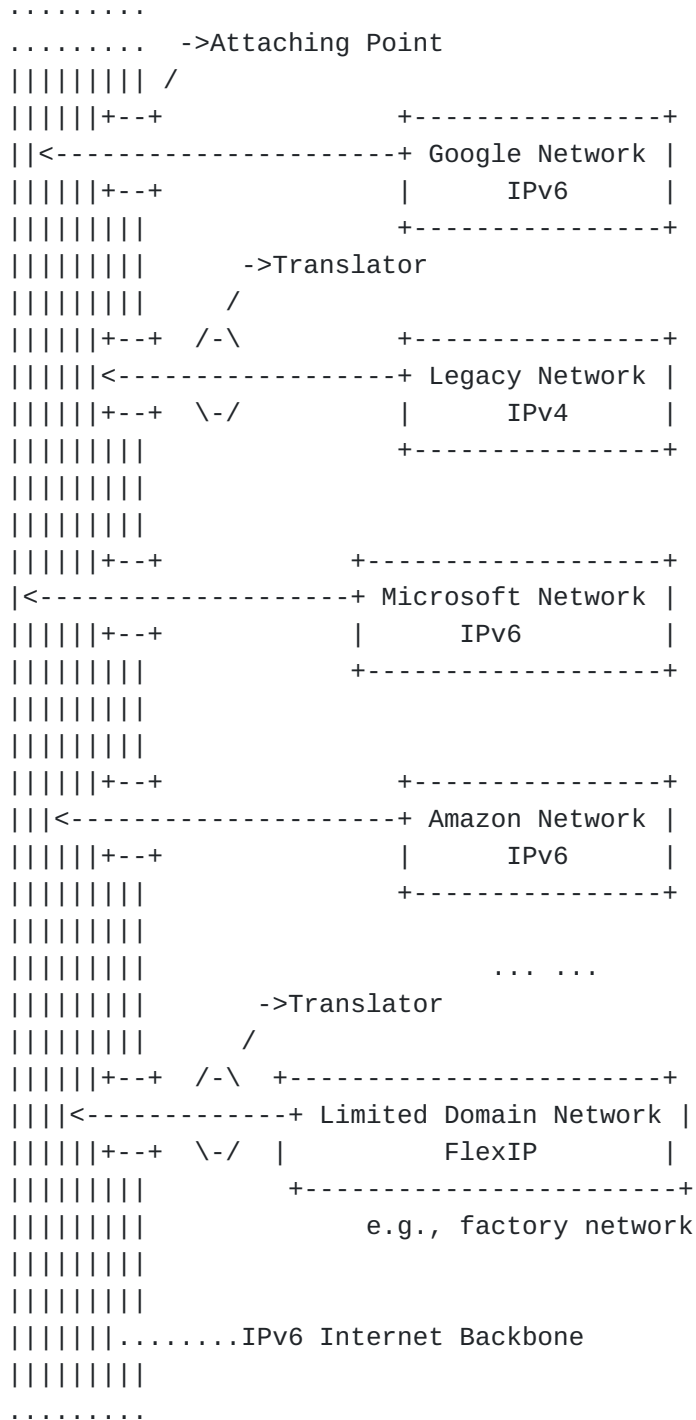


Figure 1: Position for FlexIP in IPv6 Internet

4. Design Considerations

As described in document [[draft-jia-scenarios-flexible-address-structure](#)], various address semantics and variable address length are main characteristics for a flexible IP. However, several principles must be considered from the top if such featured address

structure become truly practicable. Points below details overall considerations and plannings behind FlexIP.

4.1. Multi-Semantics

For networks with advanced routing, non-topological identifiers could be necessary for reachability [[RFC7476](#)]. To enrich routing abilities in complex network, address structure should be flexible enough to accommodate multiple semantics and related identifiers, thus forwarding nodes within the network can dealing with these complex address according based on the routing system built.

Ideally, devices that resided in advanced networks construct special address format for customized routing, while devices that resided in conventional scenario can adopt IPv6 as routine (topology semantic) address.

4.2. Elastic Address Space

The primary orientation of FlexIP is to adapt different network scale, and each network uses different address length that best fit the presumptive scale. The alterable address length refers to a flexible address space, and such resilience makes IP highly adaptable to theoretically infinite space as well as tailored space specifically for low power scenarios.

Ideally, Autonomous System (AS) that composing the Internet is constituted by numerous networks. Each of the network hold the flexible address space that best fit their scenarios.

4.3. Scalability

A constrained space is impossible to accommodate communications among volume devices, while boundless space is unsustainable due to the explosion of global routing table. To makes the flexibility truly values, FlexIP must reach a balance between rigorous requirements of expansive address space and efficient routing performance.

Ideally, the address should be expanded to boundless space, while the structure guarantees the performance of fast forwarding.

4.4. Interoperability

FlexIP network is designed to be an attached network at edges of the Internet, thus interoperability is needed between IPv6 and FlexIP. Based on the interoperability, FlexIP, on the one hand, can benefit from the advanced network abilities and adapt to new scenarios. On the other hand, global reachability from IPv6 Internet makes the network truly values and convenient for realistic use.

Ideally, a translator module is needed wherever a boundary across FlexIP networks and IPv6 networks. The translator depicted in [Figure 1](#) gives a brief architectural instance for interoperability.

5. FlexIP Address structure

In general, FlexIP is composed through a hierarchical, self-explanatory address structure. Such hierarchical, self-explanatory address structure brings FlexIP elastic address space and multi-semantics without sacrificing scalability. [Table 1](#) details the structure of the FlexIP address structure.

Index	Type	Structure (default by topology semantic and 1 segment)
0x01	Restrained Space	topology address - address 1
0x02	Restrained Space	topology address - address 2
...
0xEF	Restrained Space	topology address - address 239
0xF0	Extendable Space	followed by address with 16-bit length
0xF1	Extendable Space	followed by address with 32-bit length
0xF2	Extendable Space	followed by address with 64-bit length
0xF3	Extendable Space	followed by address with 128-bit length
0xF4	Extendable Space	followed by address with 256-bit length
0xF5	Extendable Space	followed by address with X-bit length
0xF6	Hierarchical Segments	followed by address with 2 segments
0xF7	Hierarchical Segments	followed by address with 3 segments
0xF8	Hierarchical Segments	followed by address with Y segments
0xF9	Multi-Semantics	followed by Non-topological semantic address
0xFA - 0xFF	None	reserved

Table 1: Flexible IP Address Structure

Shapes in [Table 1](#) describes the formal representation of FlexIP and should be used in programing. Text representation of FlexIP is described in [Section 6](#). Particularly, formal representation of FlexIP in this document introduces "/" for readability only. Such "/" **MUST** be omitted in practical use.

5.1. Restrained Space Format

The first address format is called restrained space format and specific for small address space. Such format includes a 1-byte

space customized for constrained resource devices. Structure in such format guarantees that within FlexIP structure, devices can reach the shortest address length under variable length structure and pursue the maximal routing efficiency.

5.2. Extendable Space Format

The second address format is called extendable space format. By adopting such format, administrator can choose address length that best fit the network.

Specifically, for networks larger than 239 address space, a 16-bit, 32-bit, 64-bit, 128-bit, and 256-bit can be used by the network with Index F0-F4, respectively, and then followed by address itself. Particular, a IPv6 (128-bit) address is regarded as a special indexed by Index F3.

For networks prefer a customized space, a 1-byte LenIndex is emerged between Index and the address. Structure in such format ensures that address space becomes theoretically elastic and boundless. For example, a 56-bit address is presented by F5/07/3B3A297F50C24F under FlexIP structure. Sequence value 07 refers to the 7-byte (56-bit) address length.

5.3. Hierarchical Segments Format

The third address format is called hierarchical segments format. By adopting such format, an FlexIP address is composed by multiple segments. Logically, a segment inside the address could be considered as an individual routing identifier. Thus within different routing areas, routers on the path should forward packets based on the exclusive segment. The partitioning of the address logically splits the large address into several routing segment, and segments are regarded small enough that packets flowing in separate networks could be forwarded efficiently according to the segments. For this reason, structure in such hierarchy format ensures the practicability with boundless space.

Taking an 2-segment address as an example, FlexIP F6/C8/F1/20C12AF2 present a 8-bit segment C8 and a 32-bit segment 20C12AF2, where index F6 indicates the 2 segments behind.

5.4. Multi-Semantics Format

The forth address format is called multi-semantics format. For address adopting such format, networks can forward packets based on the specific semantics.

Under such format, a 1 byte SemIndex is used as the indication of semantic when Index equal to F9. Taking the satellite network

[[space-routing](#)] as an example, FlexIP F9/01/F2/A32F84C981002E9B can refer to a geographic position embedded address, where 01 refers to the geographic semantic, and F2 refers to a 64-bit address length. Similarly, such pattern can be used for name based routing [[ndn](#)], user based routing, or service based routing.

Given that non-topological semantics and addressing are still under open study, specifications for non-topological semantics will be depicted in independent documents when technics become mature.

6. FlexIP Address Text Representation

Literally, text representation of FlexIP should be human friendly compared to the formal representation in [Section 5](#). Considering text representation would be used in extensive written places, FlexIP is such representation should be eminently readable.

This document **RECOMMENDED** text representation of FlexIP through following structure: [Length]<SEM>Value[Length]<SEM>Value... [Length]<SEM>Value. Generally, FlexIP address is concatenated directly by multiple segments. For each of the segment, the text representation is composed by [Length]<SEM>Value. Specifically, i.e., for components inside an segment, [Length] refers to the length of current segment with the Byte unit; Then followed by <SEM>, <SEM> refers to the semantic the segment use. Within a segment, <SEM> is the only field can be omitted if segment points to the default topology semantic -- <TOPO>. Last, Value refers to the address itself. Particularly, Value inherits the same text representation as IPv6 that recommended in [[RFC5952](#)].

[Table 2](#) depicts examples for FlexIP representation in text shape. Noted that "/" in the formal representation is for readability only and **MUST** be removed in practice.

Formal Representation	Text Representation
C8	[1]C8
F1/2A00012F	[4]2A::12F
F5/07/3B3A297F50C24F	[7]3B:3A29:7F50:C24F
F6/C8/F2/20010000000012F	[1]C8[8]2001::12F
F8/04/F0/2F5B/F0/6A3C/F0/9C2B/F0/735D	[2]2F5B[2]6A3C[2]9C2B[2]735D
F9/01/F2/A32F84C981002E9B	[8]<GE0>A32F84C981002E9B

Table 2: Examples of Flexible IP Address Text Representation

For example, [1]C8[8]2001::12F indicates two segments concatenation: segment C8 with <TOPO> semantic and 1 byte length, and segment 20010000000012F with <TOPO> semantic and 8 byte length. Particular, given that non-topological semantics and addressing are still under

open study, <SEM> identification should be officially maintained in IANA.

7. Interoperability

To enable global reachability and inter connectivity between FlexIP network and IPv6 network, an translator is needed wherever packets across the periphery. [Figure 2](#) depicts the core component of the translator, i.e., address mapper, and a sketch for packet traversing from a FlexIP network to a IPv6 network. For any packet leave FlexIP network and enter IPv6 network, the IP addresses of the packet should be converted by rules configured in the mapper, and vice versa.

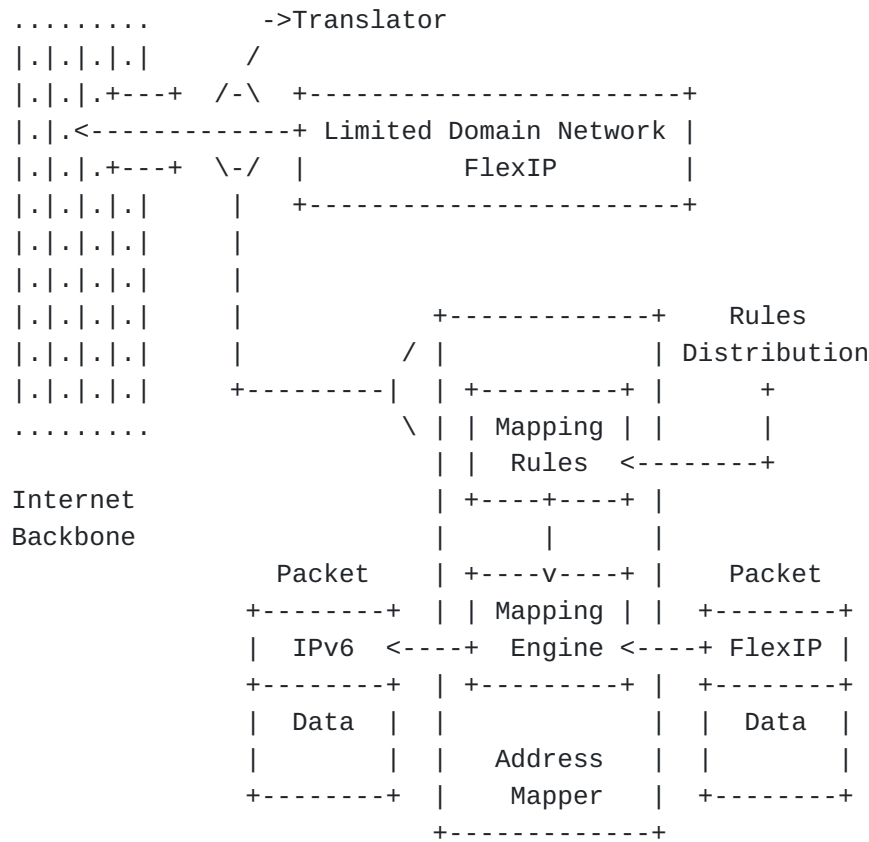


Figure 2: Network Address Mapping between FlexIP network and IPv6 network.

Specifically, there are two kind of mapping policy: stateful recording and stateless transforming. Although both two policy is effective for address mapping, stateless transforming is **RECOMMENDED** for system efficiency and operation complexity. Concrete processes of rules generation, distribution and mapping mechanisms are outside

the scope of this specification and should be documented individually.

For FlexIP network with restrained space format [Table 1](#), for instance, a FlexIP C3 should be mapped into IPv6 2001::C3 when affiliated packet flow across the periphery. Conversely, address mapper can simply peel of the prefix 2001:: of when packets flow back to FlexIP network.

8. Security Considerations

As a address format of IP, FlexIP address itself do not involve security issues. While from the viewpoint of the transmission of packets, FlexIP has security properties that are similar to IPv6. These security issues include:

- *Eavesdropping, where on-path elements can observe the whole packet (including both contents and metadata) of each datagram.
- *Replay, where the attacker records a sequence of packets off of the wire and plays them back to the party that originally received them.
- *Packet insertion, where the attacker forges a packet with some chosen set of properties and injects it into the network.
- *Packet deletion, where the attacker removes a packet from the wire.
- *Packet modification, where the attacker removes a packet from the wire, modifies it, and reinjects it into the network.
- *Man-in-the-middle (MITM) attacks, where the attacker subverts the communication stream in order to pose as the sender to receiver and the receiver to the sender.
- *Denial-of-service (DoS) attacks, where the attacker sends large amounts of legitimate traffic to a destination to overwhelm it.

Specifically, there is not any mechanism for FlexIP to protect against IP spoofing. Defending against these type of attacks is outside the scope of this specification.

9. IANA Considerations

This document does not include an IANA request.

10. Informative References

[RFC0791]

Postel, J., "Internet Protocol", STD 5, RFC 791, DOI 10.17487/RFC0791, September 1981, <<https://www.rfc-editor.org/info/rfc791>>.

[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>>.

[RFC4944] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, DOI 10.17487/RFC4944, September 2007, <<https://www.rfc-editor.org/info/rfc4944>>.

[RFC5952] Kawamura, S. and M. Kawashima, "A Recommendation for IPv6 Address Text Representation", RFC 5952, DOI 10.17487/RFC5952, August 2010, <<https://www.rfc-editor.org/info/rfc5952>>.

[RFC7476] Pentikousis, K., Ed., Ohlman, B., Corujo, D., Boggia, G., Tyson, G., Davies, E., Molinaro, A., and S. Eum, "Information-Centric Networking: Baseline Scenarios", RFC 7476, DOI 10.17487/RFC7476, March 2015, <<https://www.rfc-editor.org/info/rfc7476>>.

[RFC7668] Nieminen, J., Savolainen, T., Isomaki, M., Patil, B., Shelby, Z., and C. Gomez, "IPv6 over BLUETOOTH(R) Low Energy", RFC 7668, DOI 10.17487/RFC7668, October 2015, <<https://www.rfc-editor.org/info/rfc7668>>.

[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>>.

[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>>.

[RFC8724] Minaburo, A., Toutain, L., Gomez, C., Barthel, D., and JC. Zúñiga, "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>>.

[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>>.

[draft-jia-scenarios-flexible-address-structure]

Jia, Y., Li, G., and S. Jiang, "Scenarios and Requirements for Flexible Address structure", October 2020, <<https://xml2rfc.ietf.org/public/rfc/bibxml-ids/reference.I-D.draft-jia-scenarios-flexible-address-structure.xml>>.

[iot]

Jara, AJ., Ladid, L., and AF. Gomez-Skarmeta, "The Internet of Everything through IPV6: An Analysis of Challenges, Solutions and Opportunities", Networks Ubiquitous Comput. Dependable Appl. 4(3): 97-118, 2013.

[ndn]

Zhang, L., Afanasyev, A., and J. Burke, "Named data networking", ACM SIGCOMM Computer Communication Review 44(3): 66-73, 2014.

[space-routing]

Yang, Z., Li, H., Wu, Q., and J. Wu, "Analyzing and optimizing BGP stability in future space-based internet", International Performance Computing and Communications Conference (IPCCC) , December 2017.

[thread]

Thread Group, "Thread Specification", <<https://www.threadgroup.org/ThreadSpec>>.

[waist]

Akhshabi, S. and C. Dovrolis, "The Evolution of Layered Protocol Stacks Leads to an Hourglass-shaped Architecture", Proceedings of the ACM SIGCOMM 2011 Conference 206-217, October 2011, <<https://dl.acm.org/doi/abs/10.1145/2018436.2018460>>.

Authors' Addresses

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

Email: jiayihao@huawei.com

Zhe Chen
Huawei Technologies Co., Ltd
156 Beiqing Rd.
Haidian, Beijing
100095
P.R. China

Email: chenzhe17@huawei.com

Sheng Jiang
Huawei Technologies Co., Ltd
156 Beiqing Rd.
Haidian, Beijing
100095
P.R. China

Email: jiangsheng@huawei.com