6lo Working Group
Internet-Draft
Intended status: Standards Track
Expires: 9 September 2023

L. Iannone, Ed.
G. Li
D. Lou
Huawei
P. Liu
R. Long
China Mobile
K. Makhijani
Futurewei
P. Thubert
Cisco

8 March 2023

Path-Aware Semantic Addressing (PASA) for Low power and Lossy Networks
draft-ietf-6lo-path-aware-semantic-addressing-00

Abstract

This document specifies a topological addressing scheme, Path-Aware Semantic Addressing (PASA) that enables IP packet stateless forwarding.
No routing table needs to be built, rather, the forwarding decision is based solely on the destination address structure. This document focuses on carrying IP packets across an LLN (Low power and Lossy Network), in which the topology is static, the location of the nodes is fixed, and the connection between the nodes is also rather stable. This specifications describes the PASA architecture, along with PASA address allocation, forwarding mechanism, header format design, and IPv6 interconnection support.

Copyright Notice

Table of Contents

1. Introduction

There is an ongoing massive expansion of the network edge, driven by the "Internet of Things" (IoT), especially over low-power links which often, in the past, did not support IP packet transmission.

Particularly driven by the requirements stemming from Industry 4.0, Smart Grid and Smart City deployments, more and more devices/things are connected to the Internet. Sensors in plants/parking bays/mines/data-centers, temperature/humidity/flash sensors in buildings/museums, normally are located in a fixed position and are networked by low power and lossy links even in hardwired networks. Comparing with traditional scenarios, scalability of the (edge) network along with lower power consumption are key technical requirements. Moreover, large-scale Low power Lossy Networks (LLNs) are expected to be able to carry IPv6 packets over their links, together with an efficient access to native IPv6 domains.

The work in [SIXLOWPAN]/[SIXLO]/[LPWAN] Working Groups addresses many fundamental issues for those type of deployments, which can be considered an instantiation of what [RFC8799] defines as "limited domains". For instance, the 6lowpan compression ([RFC4944], [RFC6282]) addresses the problem of IPv6 transmission over LLNs, making it possible to interconnect IPv6-based IoT networks and the Internet. [RFC8138] introduces a framework for implementing multi-hop routing on an LLN using a compressed routing header, which works also with RPL (Routing Protocol for LLNs [RFC6550]). This technique enables the ability to forward IPv6 packets within the domain without the need of decompression. In addition, SCHC (Generic Framework for Static Context Header Compression and Fragmentation [RFC8724]) enables even more compression by using a common stateful static context.

The aforementioned technologies, which leverage on the presence of a routing protocol, are suitable in generic IoT scenarios and LLN networks. The above technologies leverage topology discovery or routing mechanisms, whereas there are several special-purpose networks, where routing protocols are not deployed and the networks are statically manageable [I-D.ietf-6lo-use-cases] (e.g. PLC [I-D.ietf-6lo-plc] or MS/TP [RFC8163], and Industrial IoT technologies like [RS485], etc.). In those kinds of deployments, topologies are planned in advance and well provisioned, with sensor nodes usually in fixed locations. This document introduces a topology-based addressing mechanism with that allows to avoid the use of routing protocol in favor of a topological stateless forwarding algorithm (see Section 3).

This specification document leverages on the 6Lo Routing Header (6LoRH) as defined in [RFC8138] and LOWPAN_IPHC header compression [RFC6282]. The use of other compression techniques is out of the scope of this document, and may be the object of separate specifications. The proposed addressing is independent of Unique Local Addresses [RFC4193], which has a dependency on specific link-layer conventions [RFC6282]. It is also different from stateful address allocation that requires all nodes to obtain addresses from a centralized DHCP server, which leads to increased network startup time and consumption of extra bandwidth. Compared to RPL-based routing [RFC6550], PASA avoids the extra overhead of address assignment by integrating address assignment and tree forming together. Furthermore, PASA provides much smaller forwarding table size than storing mode RPL.

2. Requirements Notation

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] and [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Comprehensive Use Cases

As mentioned in Section 1, the [I-D.ietf-6lo-use-cases] provides some 6lo use cases with wired connectivity, tree-based topology, and no mobility requirement (cf. Table 2 of [I-D.ietf-6lo-use-cases]). These use cases, where PASA can be used, include Smart Grid, Smart Building, etc. The PASA solution utilizes stable and static topology information to allocate addresses for nodes, which enables stateless forwarding. It saves overhead of messages triggered by routing protocols and reduces RAM footprint for routing table storage. Thus, it will reduce the overall energy consumption. The PASA forwarding logic is extremely simple, few lines of code are sufficient to implement the stack. It enables the solution being ported onto very constrained nodes. In the following paragraphs, we will dive deeper into a few use cases to demo the applicability of the PASA solution.

3.1. Smart Grid

A typical smart grid network topology whose purpose is to distribute electricity to homes in a residential area consists of Smart Circuit Breaker (SCB), Phase Change Switch (PCS), Cable Branch Box (CBB) and Power Distribution Cabinet (PDC), as shown in Figure 1. The PDC containing a few SCBs, phase compensation units, sensors and actuators is responsible for the power distribution towards CBB. The CBB containing SCBs and sensors further distributes the power to PCS

and eventually to the home. The smart grid power distribution network forms a typical tree topology, where the PLC communication technology is used to collect data (meter numbers, phases, etc.) and perform control/management of the overall system.

```
                             +---Voltage Transformer
                             |
                  +----------+-----------+
                  | PDC    +-+-+         |   SCB:Smart Circuit Breaker
                  |        |SCB|         |   PCS:Phase Change Switch
                  |        +-+-+         |   CBB:Cable Branch Box
                  |   +------+-------+   |   PDC:Power Distribution
                  | +-+-+  +-+-+   +-+-+ |       Cabinet
                  | |SCB|  |SCB|   |SCB| |
                  | +-+-+  +-+-+   +-+-+ |
                  +-+---------+-------+--+
                   /          |       +-------------------------+
                  /           +----------+                      |
                 /                       |                      |
    +-----------+----------+ +-----------+----------+           |
    | CBB       |          | | CBB       |          |  Chargers |
    |   +-------+------+   | |   +-------+------+   |      ++   |
    | +-+-+   +-+-+  +-+-+ | | +-+-+   +-+-+  +-+-+ |      ||---+
    | |SCB|   |SCB|  |SCB| | | |SCB|   |SCB|  |SCB| |      ++   |
    | +-+-+   +-+-+  +-+-+ | | +-+-+   +-+-+  +-+-+ |      ++   |
    +---+-------+------+---+ +---+-------+------+---+      ||---+
       |        |      |         |       |      |          ++   |
       |        |      |       +-++    +-++ +--++
     +-+-+    +-+-+  +-+-+     +--+    +--+ +--+|
     |PCS|    |PCS|  |PCS|     Monitors for end |
     +---+    +---+  +---+                      |
                                     +CBB-------+----------+
                                     |  +-------+-------+  |
                                     |+-+-+   +-+-+   +-+-+|
                                     ||SCB|   |SCB|   |SCB||
                                     |+---+   +---+   +---+|
                                     +---------------------+
```

Figure 1: The topology of smart grid.

3.2. Smart Home

Smart home or home domotica is another example, as shown in figure Figure 2, where a PLC router (PLC-R) in each room is used to connect home appliances (boiler, dishwasher, fridge, etc.) and devices (lights, doorbell, sound boxes, etc.) to home network and sometimes to the Internet. The network can be further extended if a switch/ router is connected. As it leverages the power line distribution,

the network forms a typical tree topology as well. Some observations and considerations are:

* Usually a Home Gateway bridges the smart home to the Internet.

* The Home Gateway, the PLC routers, and most of the home appliance are fixed in different locations. They rarely move after setup.

* The smart home automation requires any to any communication.

* Lightweight communication stack with limited MCU and RAM consumption is desired.

```
                            /----------\
                           |  Internet  |
                            \-----+----/
                                  |
                           +------+------+
                           | Home Gateway|
                           +------+------+
                                  |
                   +--------------+----------------------+
+------------------|----++--------|-----------++---------|---------+
|                  |    ||        |           ||         |  Kitchen|
|  Living       +--+---+||    +---+--+ Bedroom||     +---+--+      |
|  Room         |PLC-R |||    |PLC-R |        ||     |PLC-R |      |
|               +---+--+||    +--+---+        ||     +---+--+      |
|                   |   ||       |            ||         |         |
|  +-----+-----+----+   ||  +----+--+------+  ||  +------+------+  |
|  |     |     |    |   ||  |       |      |  ||  |      |      |  |
|  |     |     |    |   ||  |       |      |  ||  |      |      |  |
| /+\   /+\   /+\  /+\  || /+\     /+\    /+\ || /+\    /+\    /+\ |
||   | |   | |   ||   | |||   |   |   |  |   ||||   |  |   |  |   ||
| \-/   \-/   \-/  \-/  || \-/     \-/    \-/ || \-/    \-/    \-/ |
|      Switches    Door ||Strip  Voice   Sound||Boiler Fridge  Dish|
|Light     door bell    ||Light  Command Boxes||            Washer|
+-----------------------+|       Device       |+-------------------+
                         +--------------------+
```

Figure 2: The topology of smart home.

3.3. Data Center Monitoring

Data centers is a significant infrastructure, which requires numerous safeguards in place to protect hardware assets against cyber-attacks. Besides, environmental issues such as extreme temperature, high humidity, water leakage and high dust concentration can cause device failures as well. Therefore, it is critical to deploy sensors to

monitor environmental factors to make sure data center is running efficiently.

The network topology of the data center supervision system is hierarchical, and mainly consists of Network Management System (NMS), Supervision Center (SC), Field Supervision Unit (FSU), dumb and smart devices, as shown in theFigure 3. The smart devices refer to smart air conditioner, smart door lock and power equipment with embedded sensors to report their working status. The dumb devices refer to the many devices without embedded sensors, which require additional sensors to collect and update information of environment.

```
NMS:Network Management System  /----\             //------\\
SC :Supervisor Center         /      \          ||          ||
FSU:Field Supervisor Unit    |   SC   +---------+|    NMS   ||
                              \      /            \\------//
                               \----/
                              /      \
                            /         \
                        /----\         \
                       /      \          \
                      |   SC   |          \
                       \      /            \
                        \--X-/              \
                       /     \               \
                     /         \               \
                   /             \              \
                /-/-\           /-\-\           /---\
               | FSU |         | FSU |         | FSU |
                \-X-/           \-X-/           \-X-/
                /   \           /   \           /   \
               /     \         /     \         /     \
            +---+   /--\    +---+   /--\    +---+   /--\
            |   |  |    |   |   |  |    |   |   |  |    |
            |   |  |    |   |   |  |    |   |   |  |    |
            +---+   \--/    +---+   \--/    +---+   \--/
            Smart   dumb    Smart   dumb    Smart   dumb
           Device  Device  Device  Device  Device  Device
```

Figure 3: The topology of Power & Environment Supervisor System.

Both dumb and smart devices are connected to the FSU, which monitors and connects all devices of the whole floor. The number of ports on FSU is limited, where one FSU usually contains 8 analog input ports, 16 digital input ports, 4 digital output ports, 8 RS485 ports and 4 IP ports. The terminal devices report working status and environmental information to FSUs every 3 second. If values that are abnormal or above a certain threshold are detected, the FSU reports

it to the SC immediately and keeps on reporting it in real-time for next couple of hours, until the manager issues new commands. The SC can be constructed as required. The FSU reports to the local SC first, then relay the message to the central SC for data analyzing and management.

In this scenario, deployed devices (usually 600-1000 sensors per floor), due to the shortage of ports and limitation of voltage supply, use additional power supply or batteries. Since battery replacement and maintenance is costly, it is desired to have low energy consumption for longer service life. We should not only reduce the power consumption on the device level, but also on the data transmission level. The data transmission also causes huge power consumption, which can be reduced by leveraging low power transmission protocol. The FSU connects to sensors with wired technology, such as AI/DI/RS232/RS485/single pair ethernet. Multiple FSUs will connect to hierarchical supervision centers and then make data communication with supervision platform by IPv6.

3.4. Industrial Operational Technology Networks

The Operational Technology (OT) networks are not pure IP networks. Shop floors deploy fieldbus protocols such as Modbus, Profinet/IP, BacNET, CAN etc. for process control using field devices (sensors and actuators). To improve automation, Industry 4.0 is looking at means to integrate process control in OT domain with the applications residing in IPv6 domains (the enterprise networks). This leads to three primary requirements:

* Continuity in connectivity between the end devices and applications, both of which follow different address structures.

* The OT networks are traditionally designed as layer-2 and OT operators are not expected to deploy or maintain IT style routing infrastructure, hence auto-configuration mechanisms for device addresses and reachability are preferred.

* The OT networks are also delay-intolerant; therefore, compact and lean message structures are favored over encapsulations to minimize processing and translation overheads.

Using PASA, as described in details later in this document, the following applies:

* The OT network is represented as PASA domain, interfacing with native IPv6 applications, e.g., Human-Machine Interface (HMI), Manufacturing Execution System (MES). In general on shop floors,

devices are at fixed locations or cell-sites and the PASA tree hierarchy described in Figure 4 applies suitably.

* In an idealized PASA-based OT domain, a leaf-node could be a field device (sensor or actuator) that always connects to PLC serving as last node forwarding traffic to/from the leaves, i.e. sensors and actuators.

* The border node may be at the root for any IT application requirement. Then the packet communication inside the PASA domain will strictly follow PASA structure whereas communications with IPv6 domain networks will use the Border router for translations.

```
                   IPV6   +------+------------+
          +---------------| HMI/MES/FW/Gateway|----------+
          | PASA          +------+------------+          |
          |                                              |
     +----|------------------+                           | PASA
     | +--+---+              |                           |
     | |PLC   |--------------+-----------+               |
     | +---+--+              |           |      +--------+---------+
     |    | profinet         |  +--------+----+ |        |         |
     |    |                  |  |   +-----+   | |     +--+--+      |
     |  +-----+-----+----+   |  |   | PLC |   | |     | PLC |      |
     |  |     |     |    |   |  |   +--+--+   | |     +--+--+      |
     |  |     |     |    |   |  |      |      | |        |         |
     | /+\   /+\   /+\  /+\  |  | profi|bus   | |        |  modbus |
     | \-/   \-/   \-/  \-/  |  |  +---+---+  | |  +-----+------+  |
     |  sensors/actuators    |  |  |       |  | |  |     |      |  |
     | cell-site-A           |  |  |       |  | |  |     |      |  |
     +-----------------------+  | /+\     /+\ | | /+\   /+\    /+\ |
                                | \-/     \-/ | | \-/   \-/    \-/ |
                                |             | | sensors/actuators|
                                | cell site B | | cell site C      |
                                +-------------+ +------------------+
```

Figure 4: Industrial Operational Technology Network topology.

4. Architectural Overview

Path-Aware Semantic Addressing (PASA) is an efficient topology-based network layer address assignment and packet forwarding mechanism. Each PASA node is aware of its own IPv6 address, constructed by IPv6 prefix and the PASA itself (see Section 5.1). Inside the PASA domain, nodes communicate with each other by using only PASA addresses. It is a smaller addressing space compared to the huge IPv6 addressing space, but enabling stateless forwarding using the PASA-6LoRH header (see Section 6). When IPv6 communication occurs

between nodes inside the PASA domain and external IPv6 nodes, the border router, which plays as well the role of "root" in the addressing tree, performs packet decompression (as per Section 7.2 and [RFC6282]). Note that packets destined outside the PASA domain do not need to use the PASA-6LoRh header, since they can be easily forwarded to the root following the default gateway (see Section 7.2). However, an IP-in-IP header, as for [RFC8138], is used to avoid compression/decompression at each hop. The architecture of PASA network is shown in Figure 5.

```
               /|\               Internet (IPv6)
                |               --------+--------
  IPv6 Domain   |                       |
                |                       |
                |               +-------+-------+
----------------------------    | Border Router |
                |               |  (PASA Root)  |
                |               +---------------+
                |
                |                         O
                |
                |                O         O   O
                |                  O  O
                |             O                    O
  PASA Domain    |                           O
                |          O      O  O          O    O   O
                |               O
                |                      O   O
                |                                  O
                |               O
                |
               \|/          Low-Power and Lossy Network
```

Figure 5: The architecture of general PASA networks.

In the PASA network, there are 3 types of nodes, the PASA Root, the PASA Router and the PASA Host. There is typically only one root node in the PASA network.

* PASA Root: The root node is the router responsible for the management of the whole PASA network and routing/forwarding both internal and external traffic. It uses the address Allocation Function (AF) and performs the address assignment for its children. After successful address assignment, the root will keep the state of its direct children. The root node functions as gateway between the PASA domain and the Internet. As such it also operates the translation between LOWPAN_IPHC and IPv6 formats (cf. Section 7).

* PASA Router: A PASA Router is an internal node, different from the root, having least one child. It is basically the root of a subtree and as such it is a router forwarding traffic between its parent and its children according to the addressing. When handling a packet, if the destination is in one of its subtrees, it forwards the packet to the right child, otherwise it simply sends it to its parent.

* PASA Host: A PASA Host is a node with no children, hence a leaf. It operates as an host, since it is either destination or source of every packet it handles. If it is the source of packets, it simply sends the packets to its parent.

PASA Routers and Hosts roles can be assigned similarly to IEEE 802.15.4, which distinguishes between Full-Function Devices (FFD) and Reduced Function Devices (RFD) (cf., [ZigBee]).

The address assignment described in this document relies on the address registration mechanism described in [RFC8505] (see Section 8). Each node acquiring a PASA address firstly needs to select a parent node by choosing among the nodes that replied with a Router Advertisement (RA) after an initial Router Solicitation (RS). A "first come first served" selection policy is sufficient. Then it registers its link-local address to the selected parent, asking at the same time for a PASA address. In its reply the parent will propose an address according to the node's role (indicated in the request). The proposed address is algorithmically calculated using an Allocation Function (AF). The address assigner is the parent of the node and becomes as well the default gateway from a routing perspective (used for destinations that are not in the local PASA domain). The node will then ignore replies from other neighbors.

The overall design objective is centered on reducing the size (or completely avoid the usage) of routing/forwarding table by using a topological addressing scheme. PASA reduces the amount of information synchronization messages, so it actually reduces computation complexity during packets parsing and forwarding. As such, PASA may save communication energy in an IoT LLN network.

There are two distinct PASA features that allow PASA to be efficient, namely:

1. PASA Address allocation (see Section 5),

2. Stateless forwarding (see Section 7),

5. PASA Allocation

The basic rules of allocation include:

* Each node's address is prefixed by their parent's address.

* Routers (Root and routers) run an AF (Allocation Function) to generate its children's addresses.

* All nodes run the same AF in the same network instance.

* The maximum length of the PASA address MUST NOT exceed 64 bits.

Normally, the root role is assigned to the border router when the LLN bootstraps. An example of a possible result of a PASA deployment is shown in Figure 6.

```
                         root           +--------------------------+
                              1         | append more bits to form |
                              O ----+   | brother's address        |
                           /  |  \   \  +--------------------------+
                          /   |   \    \
                         /    |    \     \
  +-------------+      /      |     \      \
  | PASA Router | 10 /       11   110\       \  111
  +-------------+  O -        O       O        O
                 / |\ \               | \
               /   | \  \             |  \
             /     |  \    \          O   O
           /       |    \    \
       100/    1010|   101   1011   +--------------+
         O         O      O      O  |Prefix is '10'|
        /|        /|              +--------------+
       / |       / |
      O  O      O  O        +-----------+
   1001 10011 10101 101011  | PASA Host |
                            +-----------+
```

Figure 6: An example of PASA addresses allocation.

Every router node stores and maintain two indexes, one for the children that are also routers and one for the children that are hosts (starting at 0 for the first child in each role). The first index is named 'r', as of routers, and the second 'h' as for hosts. The '+' symbol indicates a concatenation operation. The b() operation indicates the binary string of '1' with length equal to its argument, for instance b(3) returns '111'. The allocation function AF(role,i) used in this document is defined as:

```
AF(role, r, h) = 'address of the node performing the function'
                  + (role == host? b(h++):b(r++))
                  + (role == host?'1':'0')
```

Where 'r' and 'h' are the indexes of respectively the routers and the hosts at this layer (starting at 0). Taking the example of the topology in Figure 6, the proposed AF works as follows.

At the top level, there are 4 children of root, two are routers and the other two are hosts. Starting from the left most node and moving to the right, the root node applies the AF as follows:

* For the first child, which is a router:

  - A('router', 0, 0) = '1'(root address) + b(0) + '0' = '1' + '' + '0' = 10

  - Index 'r' is increased by one and is now equal 1 (r=1)

* For the second child, which is a host:

  - A('host', 1, 0) = '1'(root address) + b(0) + '1' = '1' + '' + '1' = 11

  - Index 'h' is increased by one and is now equal 1 (h=1)

* For the third child, which is a router:

  - A('router', 1, 1) = '1'(root address) + b(1) + '0' = '1' + '1' + '0' = 110

  - Index 'r' is increased by one and is now equal 2 (r=2)

* For the fourth child, which is a host:

  - A('host', 2, 1) = '1'(root address) + b(1) + '1' = '1' + '1' + '1' = 111

  - Index 'h' is increased by one and is now equal 2 (h=2)

The first level addresses have now been assigned. Let's now have a look to how the node 10 (the first router child of the root) applies the same Allocation Function. Note that node 10 will use its own 'r' and 'h' indexes initialized to 0. Starting again from the left most node, node 10 applies the AF as follows:

* For the first child, which is a router:

- A('router', 0, 0) = '10'(node address) + b(0) + '0' = '10' + '' + '0' = 100

- Index 'r' is increased by one and is now equal 1 (r=1)

* For the second child, which is a host:

  - A('host', 1, 0) = '10'(node address) + b(0) + '1' = '10' + '' + '1' = 101

  - Index 'h' is increased by one and is now equal 1 (h=1)

* For the third child, which is a ruoter:

  - A('router', 1, 1) = '10'(node address) + b(1) + '0' = '10' + '1' + '0' = 1010

  - Index 'r' is increased by one and is now equal 2 (r=2)

* For the fourth child, which is a host:

  - A('host', 2, 1) = '10'(node address) + b(1) + '1' = '10' + '1' + '1' = 1011

  - Index 'h' is increased by one and is now equal 2 (h=2)

Note how the children of the same parent all have the same prefix (10 in this example) and such parent will be their default gateway. The proposed AF algorithmically assigns addresses to the different nodes without the need to know the topology in advance. However, the largest address of the network will depend on the actual topology. Indeed, the maximum length of an address with the proposed AF grows linearly at each level of the tree with the number of siblings from the same parent. Let's take again the example in Figure 6 and let's assume that the children of node 10 are all hosts, for the largest address we need 2 bits to encode the parent node prefix (10 in this case) to which we need to add a number of '1' equal to the value of the l index which is the number of hosts minus one (because the first host has index 0), in this case since there are 4 hosts, the index value is 3 and we add the '111' string, hence the address length would be 6 (2 for the prefix, 3 to encode the 4th host address, and one for the final 1 the ends all hosts' addresses). In a more formal way the maximum address length at each level can be calculated as:

```
Max_Length = length(Parent address)
             length(b(max(r,h)))
             + 1
```

Where 'r' and 'h' are the indexes counting respectively the routers and the hosts at this level.

The Allocation Function can be different from the one defined in this specifications, where all nodes know which one to use by configuration (cf. Section 9). The use of one and only one AF is allowed in a PASA domain and MUST be the same for all nodes. It is RECOMMENDED that implementations support at least the AF proposed in this document.

Different allocation functions may, for example, leverage on a priori knowledge of the topology in order to optimize the maximum address size and make it smaller. For instance, because the order of address allocation has an impact on the size, the address of children with the largest subtree should be allocated in the first place so to reduce the average address length of the whole subtree. Also, knowing the traffic in advance, or being able to have an estimation, can help to minimize the size of addresses that have a lot of traffic. This kind of optimization can be an option, the specification of optimizations is out of the scope of this document and may be defined in new Allocation Functions to be added to the "Allocation Function Registry" (see Section 9).

5.1. PASA Addresses and IPv6 Addresses

Obtaining a full IPv6 address from a PASA address is pretty straightforward. First the PASA address is concatenated to the configured IPv6 prefix. Since the length of the PASA address is smaller than or equal to 64 bits (the interface ID length in IPv6), the node needs to pad it with zeros ('0') used as most significant bits. The full IPv6 address will look like: IPv6 prefix + "000...000" + PASA (or in IPv6 notation <IPv6 Prefix>::<PASA>). This is equivalent of doing a coalescence operation as described in [RFC8138] (see as well Section 6.3). The PASA is assigned by the root or router as previously described.

PASA does not prevent the normal checksum calculation for the transport layer (namely TCP or UDP) or IPSec encapsulation. Indeed, any PASA node is aware of its full IP address, which can be used for the calculation.

5.2. Limitation of Number of Child Nodes

The maximum number of child nodes is determined by the specific AF used. IEEE 802.15.5 has explored the use of a per-branch setup, which, however, incurs scalability problems [LEE10]. PASA allocation design is more flexible and extensible than the one proposed in IEEE 802.15.5. The AF used as example in this document does not need any specific setup network by network, though it is still limited by the maximum length of addresses. For the special case of the parent connecting to huge amount of children, a variant of the proposed AF can be designed to fulfill the requirement and optimize the address allocation (as previously described).

6. The PASA-6LoRH Header

The PASA encodes path information into addresses to enable stateless forwarding. Such operation can be performed without touching the stateful forwarding procedure (based on the presence of a routing protocol like RPL), aka without modifying the 6LowPAN architecture, rather leveraging on mechanism already defined. In particular, by using the 6LowPAN Routing Header in Page 1, defined in [RFC8138], it is possible to define a new Critical 6LowPAN Routing Header Type, named PASA-6LoRH, that will be used by nodes to perform stateless PASA forwarding as described in Section 7.

6.1. PASA-6LoRH Sequence

The extension octets typical sequence for a compressed 6LowPAN packet with PASA Routing Header is shown in Figure 7, following the specification of [RFC8138].

```
+-----------+----...----+--------...------+----...----+
|  11110001 | PASA-6LoRH |   LOWPAN_IPHC    |  Payload  |
|  Page 1   |   Type 8   |                  |           |
+-----------+----...----+--------...------+----...----+
```

Figure 7: A lowPAN encapsulated IPv6 header compressed packet with PASA-6LoRH and LOWPAN_IPHC headers.

Where:

* PASA-6LoRH: is the PASA specific extension. See Section 6.2 for details.

* LOWPAN_IPHC: IPv6 compressed header according to [RFC6282].

These two fields are followed by the packet payload.

All nodes of a PASA domain MUST recognize the PASA critical 6LoWPAN Routing Header and be able to handle the packets according to these specifications. Otherwise, packets can be dropped, hence disrupting communications.

6.2. PASA-6LoRH Format

The format of the PASA-6LoRH header, is shown in Figure 8.

```
  0                                       1
  0   1   2   3   4   5   6   7   8   9   0   1   2   3   4   5
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| 1 | 0 | 0 | Rsvd  |   Size    |          6LoRH Type           |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|             Octet 1           |          Octet 2              |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
~                         ...                                   ~
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|             Octet N-1         |         Octet N               |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
            Where N = Size + 1, and 6LoRH Type = PASA
```

Figure 8: The PASA 6Lo Routing Header format.

Where:

* Reserved (Rsvd): Reserved for future use. It MUST be initialized to zero by the sender and MUST be ignored by the receiver.

* Size: indicates the length of the PASA address in octets. The length N equals Size plus 1, which indicates that the length of the PASA address in PASA-6LoRH is at least 1 octet and no more than 8 octets.

* Octet 1 .. Octet N: the PASA destination address used for forwarding purposes. See Section 7 for detailed forwarding operation. PASA addresses are aligned on the least significant bits. For instance, to encode the address b1011, which is the address of a host node since it terminates with '1', the corresponding octet would be b00001011 (or in hexadecimal: 0x0B).

6.3. PASA-6LoRH and LOWPAN_IPHC co-existence

In a PASA domain every node has to use PASA and being able to compress/uncompress PASA addresses according to this specification. The reference prefix of the PASA domain represents a context that can be used to compress addresses in accordance to [RFC6282] and decompress using the context and the coalescence procedure in

[RFC8138]. As such the simplest mode of co-existence of PASA-6LoRH with LOWPAN_IPHC is to use stateful address compression in the LOWPAN_IPHC header using the PASA context, then the PASA engine can just read the destination address from the LOWPAN_IPHC header, encoding it in the PASA_6LoRH header according to format previously described in Section 6.2. However, this mode of operation is sub-optimal because PASA-6LoRH already includes the destination address, hence, it can be completely elided from the LOWPAN_IPHC header.

For nodes sending packets, the first step is to create a compressed packet using [RFC6282], where the source PASA address is statefully compressed using the context and the destination PASA address statefully completely elided. The destination address is then encoded in the PASA-6LoRH in its shorter form.

In case where the destination address is an address outside the PASA domain, there is not need to use the the PASA-6LoRH header, since the packet just need to follow the default route until it reaches the root node (more details in Section 7.2).

The root node, when relaying a packet coming from outside the PASA domain, compresses the source address in the LOWPAN_IPHC header according to [RFC6282] specifications.

The opposite operations need to be performed on the receiving node. Since the destination address is completely elided in LOWPAN_IPHC the IID is obtained by its encapsulation, in this case the PASA-6LoRH. The full destination address, including the IID, can be obtained via a coalescence operation with the PASA prefix in the context as described in Section 4.3.1 of [RFC8138]. The source address is handled as defined in [RFC6282]. As an example, let's assume that the PASA IPv6 prefix is 2001:db8::/64, as for [RFC8138] the reference address will be 2001:db8:0:0. Let the PASA address in the PASA-6LoRH header be 111110, which in hexadecimal is 0x3E, then the complete IPv6 address is:

```
2001:db8:0:0:0:0:0:0    Reference address
                   3E   Compressed address

2001:db8:0:0:0:0:0:3E   Coalesced address
```

In compact notation the address is: 2001:db8::3E.

7. Forwarding in a PASA Network

Internal and external communications in a PASA network work slightly differently. For internal communications, among PASA endpoints, packets carry PASA destination addresses in the PASA-6LoRH Header. For external communications, the root is responsible to perform the translation between PASA addresses and IPv6 addresses. For instance, for a packet entering into the PASA domain, the root will extract the PASA of the destination from the suffix of the IPv6 address, reducing it to the smallest set of quad that can contain the address, by removing all leading octets that are just equal to 0x00. Then the root will compress the original IPv6 and transport headers according to [RFC6282] and prepend the PASA-6LoRH header according to [RFC8138].

The following details the forwarding operations for both internal and external communication. The intra-network forwarding decision depends on the specific AF used. Here we will use the AF previously introduced (see Section 5) to illustrate the forwarding procedure.

7.1. Forwarding toward a local PASA endpoint

Inner-domain packets carry a PASA destination address in the PASA-6LoRH header. More specifically the destination address field is the address of another node in the same PASA domain. As such a PASA node performs the following sequence of actions (also see Figure 9):

1. Get destination address from the PASA-6LoRH (abbreviated to DA) and the current node's address (abbreviated to CA). Go to step 2.

2. If length of DA is smaller than length of CA, send the packet to parent node, exit. Otherwise, go to step 3.

3. If length of DA equals to length of CA, go to step 4. Otherwise, go to step 5.

4. If DA and CA are the same, the packet arrived at destination, exit. Otherwise, send the packet to parent node, exit.

5. Check whether CA is equal to the prefix of DA. If yes, go to step 6. Otherwise, send the packet to parent node, exit.

6. Calculate which child is the next hop address and forward packet to it. With the AF proposed in this document, such operation is reduced to reading the DA's bits starting from the position equals to the length of CA, then skip all '1' until the first '0'

or the last bit of DA. The sub-string obtained in such a way is the address of direct child of current node.

7. If any exception happens in the above steps, drop the packet and send an ICMPv6 "No Route to Host" notification back to the source address.

```
              /*\       DA:Destination Address
             |***|      CA:Current Node's Address
              \*/
               |
      +--------+--------+
      |Parse DA from pkt|
      +--------+--------+
               |
              \|/
       +-------+------+
      /                \  yes
     | Len(DA)<Len(CA)? |-------------------------------+
      \                /                                |
       +-------+------+                                 |
               | no                                     |
              \|/                                       |
       +-------+------+           +--------------+      |
      /                \  yes    /                \  no |
     | Len(DA)=Len(CA)? |------>|     CA == DA ?   |--->+
      \                /         \                /     |
       +-------+------+           +-------+------+      |
               | no                       | yes         |
              \|/                        /*\            |
       +-------+------+                 |***|           |
      /                \  no             \*/            |
     | CA==PrefixOf(DA)?|------------------------------>+
      \                /                                |
       +-------+------+                                 |
               | yes                                    |
              \|/                                      \|/
     +---------+---------+                    +---------+---------+
     | Calculate next-hop|                    | Forward to Parent |
     |         &         |                    +---------+---------+
     |      Forward      |                              |
     +---------+---------+                              |
               |<---------------------------------------+
              \|/
              /*\
             |***|
              \*/
```

Figure 9: Flow Chart of Internal Forwarding Procedure

In the case of packets arriving from the Internet (external IPv6 domain toward the local PASA domain) header adaptation operation is performed by the root node. It first compresses the IPv6 header according to [RFC6282] and also described in Section 6.3. The root builds the PASA address of the destination by removing the prefix and the leading '0's octets of the suffix of the destination address. Then the root creates the inner-domain packet with the PASA-6LoRH header. It uses the PASA address as destination, so to route the packet as described above to the destination node.

7.2. Forwarding toward an external IPv6 address

When the packet is destined to an external IPv6 address, it is an outer-domain packet. In this case there is no need to use the PASA-6LoRH encapsulation. Indeed, since each node has a default gateway entry in the routing table, namely its parent, so all PASA nodes (except root) just send packets that are destined outside the local domain to their parent. Eventually all packets will reach the root node, which acts as border gateway.

When the network forwarding operation are based on [RFC8138], the source node encapsulates the the LOWPAN_IPHC packet with the IP-in-IP 6LoRH Header defined in Section 7 of [RFC8138]. Where the encapsulator address is always the source address in the LOWPAN_IPHC header and the destination is always implicitly the root node. The latter will decapsulate and decompress the packet. Hence, according to [RFC8138] the IP-in-IP 6LoRH will have the form depicted in Figure 10.

```
  0                   1                   2
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |1|0|1| Length  | 6LoRH Type 6  |  Hop Limit    |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 10: IP-in-IP 6LoRH in a PASA domain.

Where the Length field is set to 1 to indicate that only the Hop Limit field is present. Such a header is positioned before LOWPAN_IPHC as shown in Figure 11.

```
 +-----------+----...----+--------...------+----...----+
 |  11110001 |  IP-in-IP  |   LOWPAN_IPHC    |  Payload  |
 |  Page 1   |   6LoRH    |                  |           |
 +-----------+----...----+--------...------+----...----+
```

Figure 11: A lowPAN encapsulated IPv6 header compressed packet with IP-in-IP and LOWPAN_IPHC headers.

8. PASA Address Configuration

[RFC8505] Registration Extensions for IPv6 over 6LowPAN Neighbor Discovery can be further extended to accommodate PASA address configuration. In order for a PASA node to request an address, the Extended Address Registration Option (EARO) message is uses, exploiting two of the reserved bits. The format of the EARO message is shown in Figure 12.

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |     Type      |     Length    |    Status     |    Opaque     |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |Rsd|P|H| I |R|T|     TID       |     Registration Lifetime     |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                                               |
...            Registration Ownership Verifier (ROVR)           ...
 |                                                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 12: EARO Format.

All the fields in EARO message are defined in [RFC8505], except for bits P and H that are allocated by this document (see Section 9) and are defined as follows:

* PASA bit (P): If set, this flag indicates that the registration message is requesting or delivering a PASA address as part of the link-local address registration procedure.

* Host bit (H): If set, this flag indicates that the node is acting as a PASA Host, otherwise, it means that the node is acting as a PASA Router (cf. Section 4).

When a PASA node bootstraps, it typically does multicast a Routing Solicitation(RS) and receives one or more unicast Routing Advertisements (RA) messages from potential parents. The node can choose a parent on a "first come first served" basis and send a Neighbor Solicitation (NS) with a EARO message to register its link-local address to the selected parent. In this EARO message it will set the P bit, to indicate that it is also requesting a PASA address. It will set the H accordingly to its intended role. The parent, acting as routing registrar will process the received EARO message and act according to [RFC8505], and the corresponding EARO message

for the NA packet is generated. The NA message will carry the EARO message with the bits P and H set exactly as in the corresponding EARO message of the NS packet. If the returning status is 0, meaning "success" according to [RFC6775], the returning EARO message will carry as well the PASA address that the parent assigns to its child using the procedures described in Section 5. The PASA address is appended to the EARO message (whose length is now set to 3), so the returning format becomes the one depicted in Figure 13.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |   Length = 3  |  Status = 0   |    Opaque     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Rsd|P|H| I |R|T|     TID       |     Registration Lifetime     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Link-Local Address                       |
|                          of the child                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Proposed  PASA Address                    |
|                          of the child                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 13: NA EARO message example.

At this point, the child MUST register the PASA address to the same parent, but not using the P and H bits. This is in order to be in line with Section 5.6 of [RFC8505], requesting global unique addresses to be registered. Furthermore, the registration procedure has the nice property to confirm that the child accepted and will use the proposed address.

If the node that made the request is a router, it can start acting as a routing registrar so to allow other nodes to select it as a parent.

9. IANA Considerations

9.1. Critical 6LoWPAN Routing Header Type for PASA-6LoRH

This document requires IANA to assign one value of the "Critical 6LoWPAN Routing Header Type" registry, to be used according to the specification in this document, as shown in Table 1. [Note to RFC Editor: If IANA assign different values the authors will update the document accordingly]

Value	Description	Reference
8 (suggested)	PASA-6LoRH	[This Document]

Table 1: Critical 6LoWPAN Routing Header Type for PASA

9.2. Allocation Function Registry

This section provides guidance to the Internet Assigned Numbers Authority (IANA) regarding registration of values related to the PASA specification, in accordance with BCP 26 [RFC8126].

IANA is asked to create a registry named "Path-Aware Semantic Addressing (PASA) Parameters".

Such registry should be populated with a one octet sub registry named "Allocation Function" and used to identify the AF used in a PASA deployment. The sub registry is populated as shown in Table 2:

Value	AF Name	Reference
0x00	PASA Tree Allocation Function	[This Document]
0x01-0xFF	Un-assigned	

Table 2: Allocation Function sub-registry

Values can be assigned by IANA on a "First Come, First Served" basis according to [RFC8126].

9.3. Address Registration Option Flags

IANA is requested to add the content show in Table 3 to the existing sub-registry "Address Registration Option Flags" under "Internet Control Message Protocol version 6".

Bit	Description	Reference
2	P Flag	[This Document]
3	H Flag	[This Document]

Table 3: New Address Registration Option Flags

10. Reliability Considerations

Because PASA uses algorithmically generated addresses based on the network topology, nodes do not generate and store forwarding table entries in the normal case. One of the potential issues is the risk of renumbering of addresses in case of topology changes. Because of the applicability domain of PASA, the common case of topology change is known in advance and can be planned, so to reduce disruption due to renumbering. Another case is temporary link failures, where the underlying technology is still able to provide connectivity through alternative links, which is strictly related to the underlying technology, the network topology, the deployed redundancy, and the expected reliability.

More complex reliability scenarios and alternative solutions are beyond the scope of this document, which is focused only on the address allocation framework and stateless forwarding. Furthermore, specific reliability solutions can depend as well on the specific Allocation Function used (different from the one presented in this document). Reliability is discussed in more details in [I-D.li-6lo-pasa-reliability].

11. Security Considerations

An extended security analysis will be provided in future revision of this document. As of this point we consider that the security considerations of [RFC4944], [RFC6282], [RFC8138], and [RFC8505] apply.

Acknowledgements

This document received many comments and help from community people. Tommaso Pecorella, Esko Dijk, Dominique Barthel, Adnan Rashid, Michael Richardson, Brian Carpenter, did provide technical comments for this document. The authors would like to thank all of them.

References

Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[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>.

[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>.

[RFC6550] Winter, T., Ed., Thubert, P., Ed., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", RFC 6550, DOI 10.17487/RFC6550, March 2012, <https://www.rfc-editor.org/info/rfc6550>.

[RFC6775] Shelby, Z., Ed., Chakrabarti, S., Nordmark, E., and C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 6775, DOI 10.17487/RFC6775, November 2012, <https://www.rfc-editor.org/info/rfc6775>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC8138] Thubert, P., Ed., Bormann, C., Toutain, L., and R. Cragie, "IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Routing Header", RFC 8138, DOI 10.17487/RFC8138, April 2017, <https://www.rfc-editor.org/info/rfc8138>.

[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>.

[RFC8505] Thubert, P., Ed., Nordmark, E., Chakrabarti, S., and C. Perkins, "Registration Extensions for IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Neighbor

Discovery", RFC 8505, DOI 10.17487/RFC8505, November 2018, <https://www.rfc-editor.org/info/rfc8505>.

Informative References

[I-D.ietf-6lo-plc]
Hou, J., Liu, B. R., Hong, Y., Tang, X., and C. E. Perkins, "Transmission of IPv6 Packets over Power Line Communication (PLC) Networks", Work in Progress, Internet-Draft, draft-ietf-6lo-plc-11, 18 May 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-6lo-plc-11>.

[I-D.ietf-6lo-use-cases]
Hong, Y., Gomez, C., Choi, Y., Sangi, A. R., and S. Chakrabarti, "IPv6 over Constrained Node Networks (6lo) Applicability & Use cases", Work in Progress, Internet-Draft, draft-ietf-6lo-use-cases-14, 24 October 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-6lo-use-cases-14>.

[I-D.li-6lo-pasa-reliability]
Li, G., Lou, Z., and L. Iannone, "Reliability Considerations of Path-Aware Semantic Addressing", Work in Progress, Internet-Draft, draft-li-6lo-pasa-reliability-00, 24 October 2022, <https://datatracker.ietf.org/doc/html/draft-li-6lo-pasa-reliability-00>.

[LEE10] Lee, M., Zhang, R., Zheng, J., Ahn, G., Zhu, C., Park, T., Cho, S., Shin, C., and J. Ryu, "IEEE 802.15.5 WPAN mesh standard-low rate part: Meshing the wireless sensor networks", DOI 10.1109/jsac.2010.100902, IEEE Journal on Selected Areas in Communications vol. 28, no. 7, pp. 973-983, September 2010, <https://doi.org/10.1109/jsac.2010.100902>.

[LPWAN] "IPv6 over Low Power Wide-Area Networks (lpwan) WG", n.d., <https://datatracker.ietf.org/wg/lpwan/about/>.

[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>.

[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>.

[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>.

[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>.

[RS485] "TIA-485-A Revision of EIA-485", n.d..

[SIXLO] "IPv6 over Networks of Resource-constrained Nodes (6lo) WG", n.d., <https://datatracker.ietf.org/wg/6lo/about/>.

[SIXLOWPAN] "IPv6 over Low power WPAN (6lowpan) - Concluded WG", n.d., <https://datatracker.ietf.org/wg/6lowpan/about/>.

[ZigBee] "ZigBee Wireless Networks and Transceivers", DOI 10.1016/b978-0-7506-8393-7.x0001-5, Elsevier book, 2008, <https://doi.org/10.1016/b978-0-7506-8393-7.x0001-5>.

Authors' Addresses

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

Email: luigi.iannone@huawei.com

Guangpeng Li
Huawei Technologies
Beiqing Road, Haidian District
Beijing
100095
China

Email: liguangpeng@huawei.com

David Lou
Huawei Technologies Duesseldorf GmbH
Riesstrasse 25

80992 Munich
Germany

Email: zhe.lou@huawei.com

Peng Liu
China Mobile
No. 53, Xibianmen Inner Street, Xicheng District
Beijing
100053
China

Email: liupengyjy@chinamobile.com

Rong Long
China Mobile
No. 53, Xibianmen Inner Street, Xicheng District
Beijing
100053
China

Email: longrong@chinamobile.com

Kiran Makhijani
Futurewei
United States of America

Email: kiranm@futurewei.com

Pascal Thubert
Cisco Systems, Inc.
France

Email: pthubert@cisco.com