           DHCPv6_PD, PDP and NDP Implementation in IoT Router (DANIR)
                       draft-shytyi-v6ops-danir-03.txt

Abstract

   This document provides a description of the implementation of Dynamic
   Host Configuration Protocol version 6 Prefix Delegation, Neighbour
   Discovery Protocol and of the use of the Packet Data Protocol in an
   Internet of Things Router. This Internet of Things Router is
   connected on a cellular network; it is a DHCPv6-PD Client and it
   requests a /56 pool of prefixes from the server; the DHCPv6-PD server
   is placed in the PGW and is a part of the cellular infrastructure.
   After the pool of prefixes is delegated, the Internet of Things
   Router derives sub-prefixes from the prefix pool; each one of these
   sub-prefixes is aimed at one ingress interface.

   After the Internet of Things Router finishes the network prefix
   assignment procedure, it advertises the network prefixes on the
   ingress links by using the Neighbour Discovery protocol. Finally,
   when Hosts receive the sub-prefixes via Router Advertisement
   messages, they configure the Global Unique Address with the Stateless
   Address Auto-configuration protocol.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

   This document describes the implementation of the Dynamic Host
   Configuration Protocol vestion 6 with Prefix Delegation (DHCPv6_PD),
   Neighbour Discovery Protocol (NDP) and usage of the Packet Data
   Protocol (PDP) in an Internet of Things (IoT) Router.

   The use of DHCPv6 Prefix Delegation in LTE networks is overviewed in
   [RFC6653]. It misses several important aspects.

The router MUST be a node that forwards IP version 6 packets not
explicitly addressed to itself [RFC8200]. Thus, it has more than one
link to perform the forwarding. With multiple links, the need of
multiple global unique network prefixes (GUNPs) , assigned to those
links, appears. To assign the GUNPs to the links, the Requesting IoT
Router Solicits the pool of GUNPs.

First, the Requesting IoT Router solicits the pool of the GUNP from
the Delegating Router.

After the pool is received, the Requesting IoT Router (1) derives
GUNPs and (2) performs address autoconfiguration. During the
autoconfiguration process the Requesting IoT Router assigns the GUNPs
to the links. When the IoT Router finishes the GUNPs assignment
procedure, it starts to advertise the GUNPs on the links with NDP
[RFC4861]. Meanwhile, the Hosts that are connected to the Requesting
IoT Router run the SLAAC mechanism to perform the GUA IP version 6
autoconfiguration.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in RFC 2119 [RFC2119].

Router - is a node, that performs forwarding. Router node is not a
final destination of forwarded packets.

Delegating Router - is a node, DHCPv6 server, that chooses prefix(es)
for delegation and advertises them to the Requesting Router
[RFC3633].

Requesting IoT Router - is a node that behaves as DHCPv6 client. It
requests the network prefix(es) and assigns network prefix(es) to the
interfaces [RFC6653].

Host - is a node that is not a router.

Link - is an entity that enables link layer communication of nodes.

Interface - node connection to the link.

Link-local address - is an address with usage that is limited by a
link.

Global Unique Address (GUA) - is an address that is globally
available and globally unique.

3. Overview

This section provides an overview of the actions performed on the
Delegating Router, Requesting IoT Router and host to perform address
assignment on the interfaces with different GUNP. The process of IP
version 6 address assignment starts with advertising of the GUNP pool
from the Delegating Router to the Requesting IoT Router. To perform
such a solicitation, the Requesting IoT Router runs the DHCPv6_PD.

```
                    +-----------------+
                    |                 |
                    |    Delegating   |
                    |      router     |
                    |                 |
                    | (DHCPv6 server) |
                    +-----------------+
                            |
                            |DHCPv6_PD network prefix
                    POOL    |2001:DB8:XYZU:TVWZ::/56
                            |
                            | GUA Prefix_0 /64
                            |interface_0
              +-------------------------------+
              |                               |
              |           Requesting          |
              |           IoT router          |
              |                               |
              |(DHCPv6_PDclient + RADVDserver)|
              +-------------------------------+
              |interface_1    |interface_2    |interface_3
              |               |               |
       GUNP   |Prefix_1 /64   |Prefix_2 /64   |Prefix_3 /64
              |SLAAC GUA      |SLAAC GUA      |SLAAC GUA
        +------------+  +-----------+  +-------------+
        |            | |           | |             |
        |   Host 1   | |   Host 2  | |    Host 3   |
        |            | |           | |             |
        +------------+  +-----------+  +-------------+
```
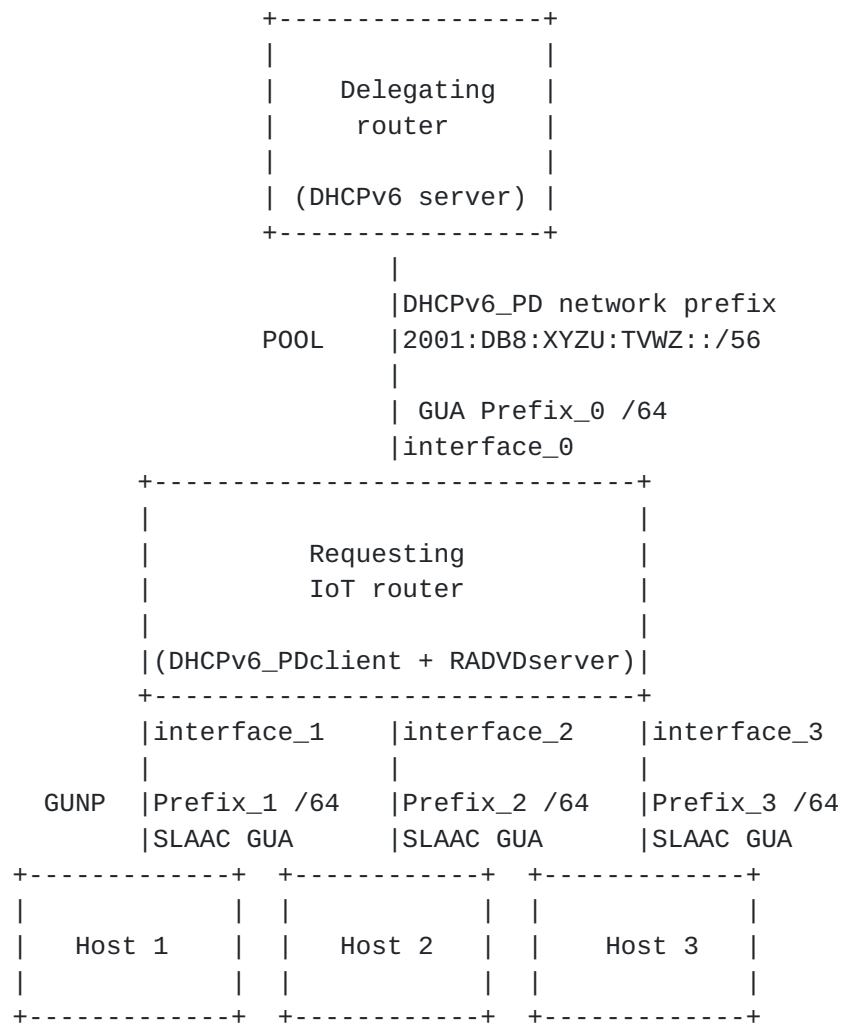
(In the above figure the scenario with 3 hosts connected to the
Requesting IoT router is presented. Normally there are no number
limitations of connected hosts.)

When the DHCPv6 message exchange is performed, the Requesting IoT
Router receives the pool of IPv6 GUNPs. After the pool of GUNPs is
received, the Requesting IoT Router performs the autoconfiguration.

Precisely, when the Requesting IoT Router's interface is attached on
the link, the Requesting IoT Router assigns to this link the GUNP
taken from GUNP pool. The Requesting IoT Router performs the GUNP
assignment procedure for multiple links over the interfaces.
Finally, this mechanism offers the automated assignment of GUNPs to
the links. At the moment of autoconfiguration, the Requesting IoT
Router's interfaces are already assigned with link local addresses.

The next step of the autoconfiguration phase SHOULD be performed
using the Neighbor Discovery protocol. The latter advertises the
network's configuration on the links with different interfaces thus
with different GUNPs. To perform the GUNPs advertisement, the
Requesting IoT ROuter sends the "Router Advertisement Messages" via
its interfaces. The Router Avertisement Messages carry the GUNPs
that are further used by the stateless autoconfiguration mechanism
(SLAAC). There exists an open source implementation of Neighbor
Discovery protocol - RADVD sever. With RADVD it is possible to
configure hosts interfaces connected to the router's interfaces in
automatic manner. It MAY be possible thanks to the fact that hosts
run the SLAAC.

The IP version 6 stateless autoconfiguration mechanism enables hosts
to perform the address autoconfiguration. The SLAAC mechanism SHOULD
be used when it is enough to have random unique IP version 6
addresses [RFC4862]. The length of the IP version 6 address is N
bytes. The first part of the address (128-N bits) consists of the
GUNP information associated with a link. The network prefix is
advertised by the IoT Router on the link. The second part of the
address (8 bytes) consists of interface identifier on the link. The
interface identifier SHOULD be generated locally and randomly.

```
+----------------------------------------+------------------------+
|              128 - N bits              |        N bits          |
+----------------------------------------+------------------------+
|              link prefix               |  interface identifier  |
+----------------------------------------+------------------------+
```

(In the above figure, an IP version 6 address scheme is presented
[RFC4862].)

3.1. Environment

This section describes the location of the Delegating Router and the
Requesting IoT Router in the cellular provider's infrastructure
model. The model is not a real cellular provider infrastructure.

After the DHCPv6 packet leaves the cellular interface of the IoT
Requesting Router via wireless OFDMA link, it reaches the element of
the access network which connects to the user equipment (UEs) -
eNodeB station.

Further, the packet MAY be transmitted to the Serving Gateway (SGW),
as all the user's IP packets. SGW is used to enable the UE movement
between eNodeBs. When the UE moves between eNodeBs, the SGW keeps
information about the bearers.

The final destination of the DHCPv6 packet is the Packet Data Network
Gateway, that is responsible for IP address allocation for the UE and
for filtering of down-link user's IP packets into the different QoS-
based bearers.

The model of the infrastructure described in this this section is a
simplified example. Real infrastructure construction could contain
multiple SGW and eNodeBs and network equipment (that is not described
in the current example) with respect of existing standards [RFC6459].

```
                   +---------------+       |
                   |      PGW      |       |
                   |   Delegating  |       |
                   |     Router    |       |
                   +---------------+       |
                        |                  |
                        |                  |
                   +---------------+       |
                   |      SGW      |       | Cellular infrastructure
                   +---------------+       |
                        |                  |
                        |                  |
                   +---------------+       |
                   |    eNodeB     |       |
                   +---------------+       |
                        |                  |
                        |                  |
                        ~                  ~  Wireless OFDMA 3G/4G
                        |                  |
                   +---------------+       |
                   |   +-------+   |       |
                   |   | Modem |   |       |
                   |   +-------+   |       |
                   |       |       |       |
                   |   +-------+   |       |
                   |   |  ARM  |   |       |
                   |   +-------+   |       |
                   |   Requesting  |       | User Equipment
                   |       IoT     |       |
                   |     Router    |       |
                   +---------------+       |
```
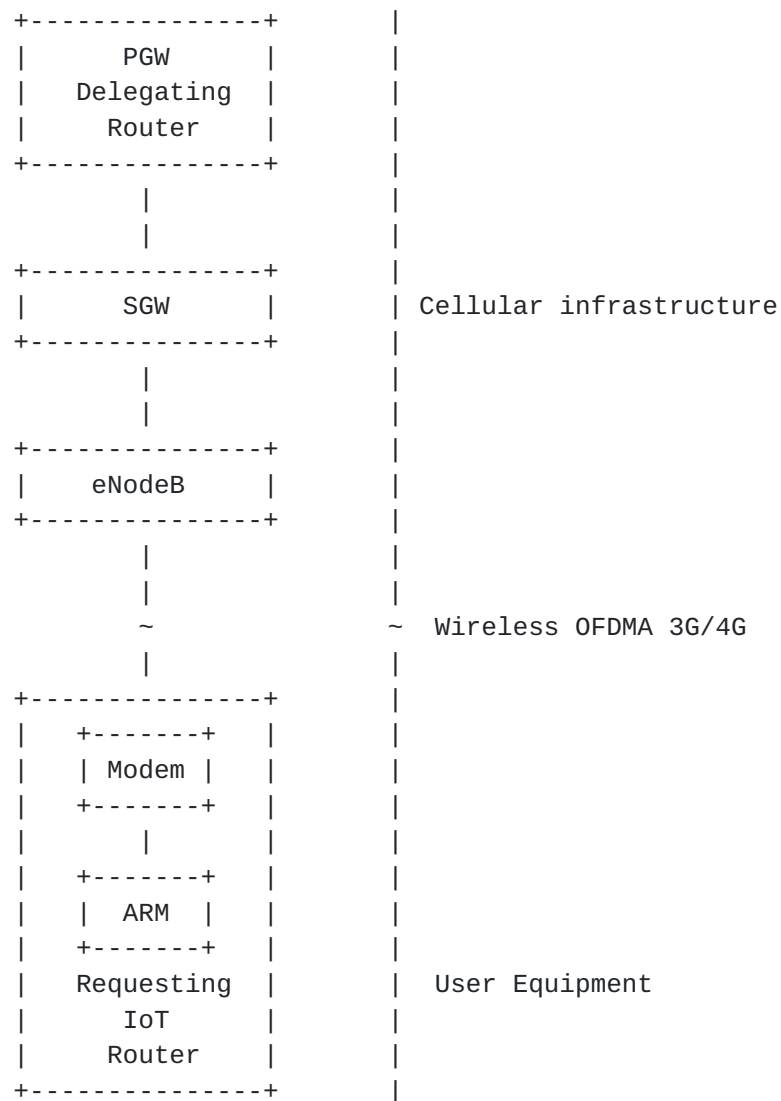
   (The above figure descibes the model of the path followed by a DHCPv6
   packet from IoT requesting router to the Delegating PGW router. The
   model is not a real infrastructure.)

   Additional experiments with using of USB dongle were performed. The
   following figure illustrates the successful DHCPv6-PD test on Orange
   with dongle. It uses a Huawei E392 USB dongle on laptop (and not the
   Sierra Wireless mangOH Red).

```
                    +---------------+       |
                    |      PGW      |       |
                    |   Delegating  |       |
                    |     Router    |       |
                    +---------------+       |
                           |               |
                           |               |
                    +---------------+       |
                    |      SGW      |       | Cellular infrastructure
                    +---------------+       |
                           |               |
                           |               |
                    +---------------+       |
                    |     eNodeB    |       |
                    +---------------+       |
                           |               |
                           |               |
                           ~               ~  Wireless OFDMA 3G/4G
                           |               |
                    +---------------+       |
                    |   +--------+  |       |
                    |   |  E392  |  |       |
                    |   |  USB   |  |       |
                    |   | dongle |  |       |
                    |   +--------+  |       |
                    |       |       |       | User Equipment
                    |   +--------+  |       |
                    |   | Laptop |  |       |
                    |   +--------+  |       |
                    |       |       |       |
                    +---------------+       |
```
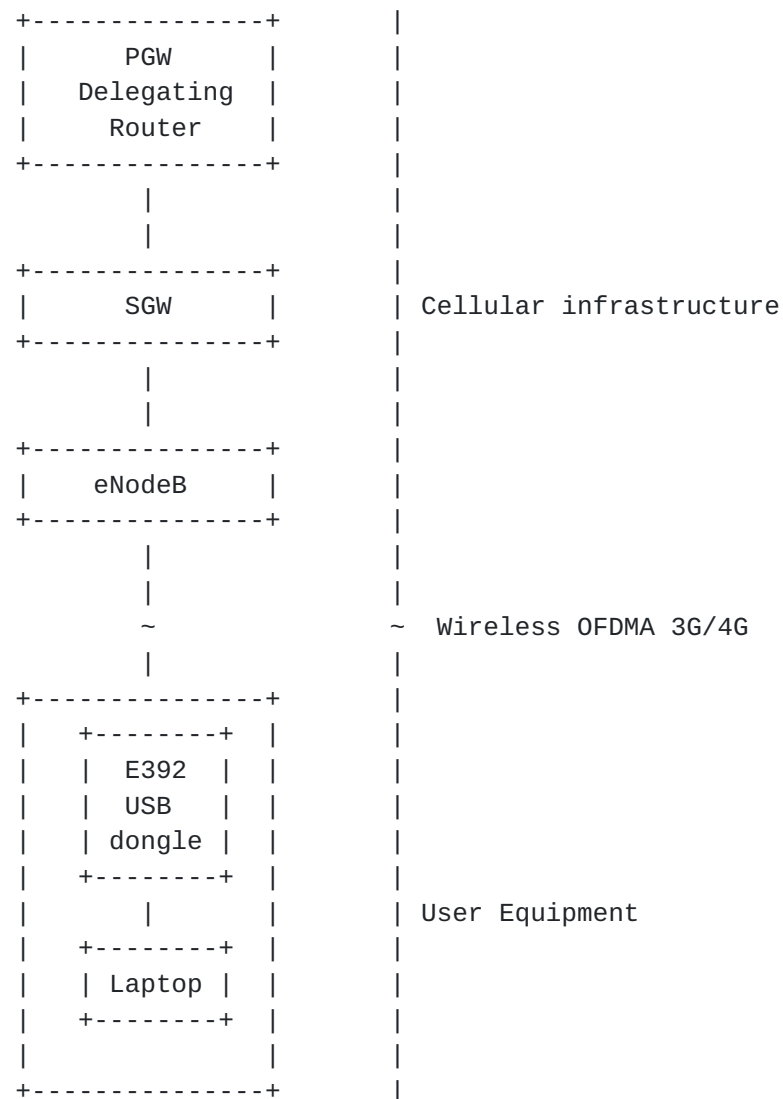
[4]. Specification

This section presents the process that starts with delegation of
Network Prefix pool 2001:DB8:XXXX:XX::/56 from the Delegation Router
and finishes with the configuration of IPv6 prefixes on the hosts
interfaces. Each Requesting IoT router's interface acts on a unique
link. The Host's interfaces, connected to the Requesting IoT router,
acts on the same unique links as the Requesting IoT router's
interfaces.

4.1. **Solicitation of the network prefix pool**

4.1.1. **The Packet Data Protocol**

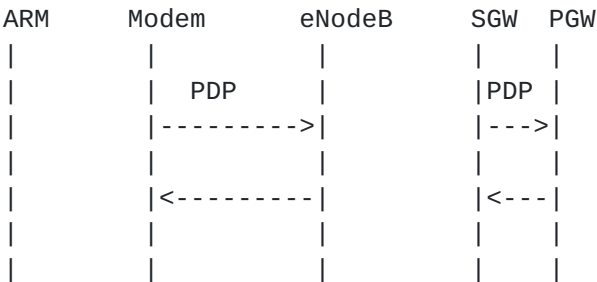   This section describes how the Requesting IoT Router obtains the GUA
   address on the Recipient Interface (RI) (OFDMA interface, 3GPP
   interface). The message "Activate PDP context Accept" is useful for
   forming the Globally Unique Address on the RI.

   The Packet Data Protocol [ETSI102361] contains the following types of
   DLL (Data Link Layer) bearer service data transmissions: unconfirmed
   data transmission; confirmed data (data transmission; response
   transmission.) The Packet Data Protocol contains the following types
   of layer 3 bearer service data transmissions: Internet Protocol;
   Short Data. These layer 3 bearer services are built on the top of
   DLL services.

```
          ARM       Modem      eNodeB      SGW  PGW
           |         |          |          |    |
           |         |  PDP     |          |PDP |
           |         |--------->|          |--->|
           |         |          |          |    |
           |         |<---------|          |<---|
           |         |          |          |    |
           |         |          |          |    |
```

4.1.2. **The Dynamic Host Configuration Protocol version 6 with Prefix
        Delegation**

   To perform the pool solicitation, the Prefix Delegation options of
   the Dynamic Host Configuration Protocol version 6 (DHCPv6) are used
   [I-D.ietf-dhc-rfc3315bis].

   The Requesting IoT Router sends the DHCPv6 "Solicit" packet to the
   Delegating Router via the wireless link. The DHCPv6 "Solicit" packet
   consists of Client Identifier, Transaction ID, Elapsed time and
   Identity Association for Prefix Delegation (IA_PD) options. The
   initial "Solicit" packet triggers the 4 message exchange, that
   finishes with the reception of the GUNPs pool by the Requesting IoT
   Router.

```
          +------------+                      +----------+
          | Requesting |                      |   PGW    |
          |    IoT     |                      |Delegating|
          |   Router   |                      |  Router  |
          +------------+                      +----------+
                |                                  |
                |                                  |
                |1 Solicit                         |
                |------------->                    |
                |                                  |
                |                     2 Advertise  |
                |                     <------------|
                |                                  |
                |3 Request                         |
                |------------->                    |
                |                                  |
                |                                  |
                |                     4 Reply      |
                |                     <------------|
                |                                  |
                |                                  |
                |                                  |
```
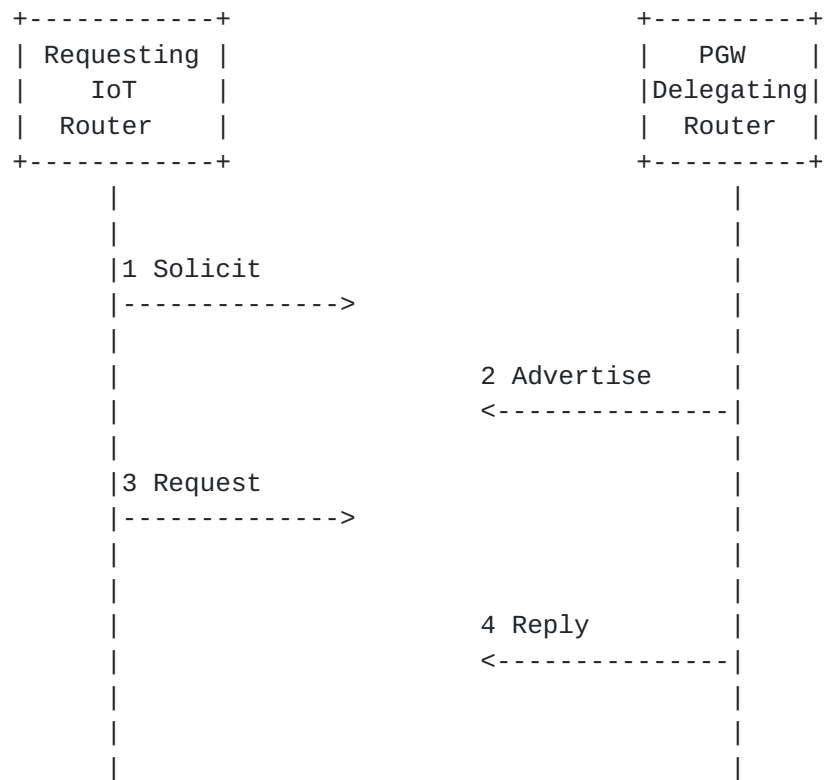
(In the above figure the full DHCPv6 message exchange mechanism
between the ARM part of UE and PGW is presented.)

The IA_PD option consists of the Identity Association (IA) - group
identifier [RFC3633], parameters (IA id, times to extend the
lifetimes of prefixes and prefixes allocated to the IA). The full
description of the IA_PD option is presented in the RFC3633
[RFC3633].

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          OPTION_IA_PD         |          option-length        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       IAID (4 octets)                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            T1                                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            T2                                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
.                                                               .
.                        IA_PD-options                          .
.                                                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
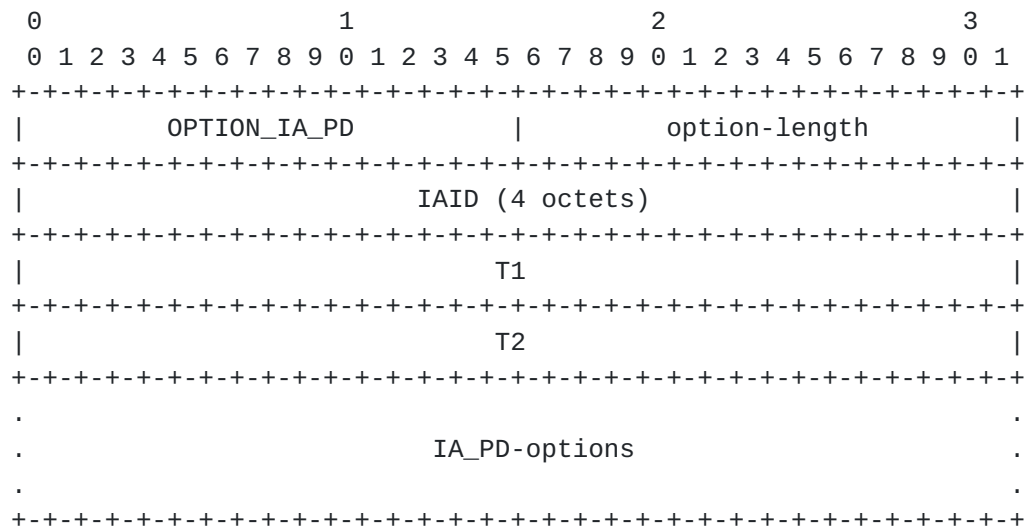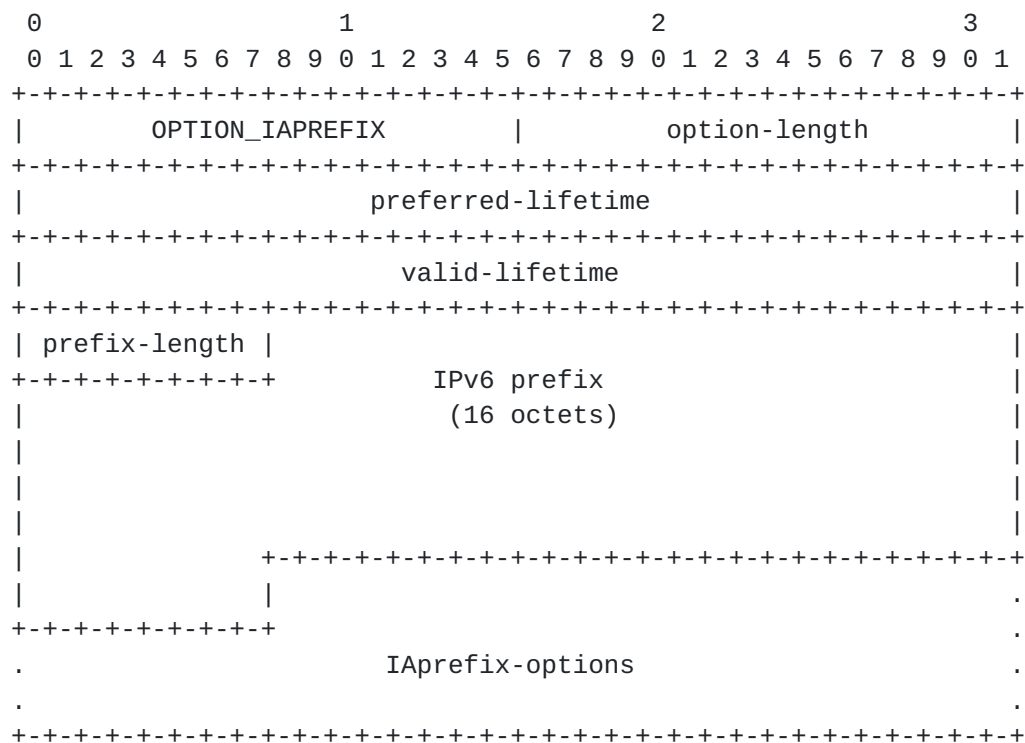
(in the above figure the DHCPv6 IA_PD option format [RFC3633].)

The IA_PD-options field carries the IA_PD Prefix Option. The IA_PD
Prefix Option carries the recommended preferred/valid life time and
IPv6 prefix with prefix length. The additional fields allocated for
the options for the advertised GUNP [RFC3633].

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         OPTION_IAPREFIX       |          option-length        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      preferred-lifetime                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        valid-lifetime                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| prefix-length |                                               |
+-+-+-+-+-+-+-+-+              IPv6 prefix                       |
|                         (16 octets)                           |
|                                                               |
|                                                               |
|                                                               |
|              +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              |                                                .
+-+-+-+-+-+-+-+-+                                               .
.                        IAprefix-options                       .
.                                                               .
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

(in the above figure the DHCPv6 IA_PD Prefix option format is
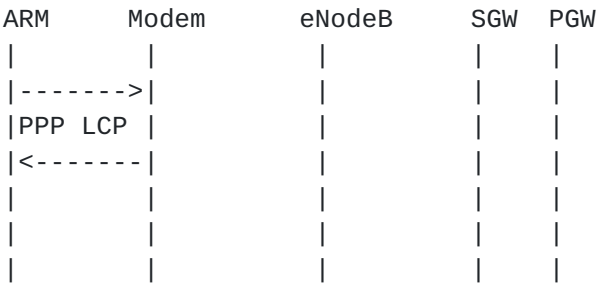presented [RFC3633].

The PGW (Delegating Router) advertises, through the usage of a DHCPv6
packet, an IPv6 pool 2001:DB8::/56 to the Requesting IoT Router. The
packet SHOULD be sent from the cellular infrastructure to the
Requesting IoT Router, via the wireless link. The full message
exchange consists of: Solicit, Advertise, Request and Reply messages.

The "Request", "Reply" messages are used to add/remove/update the
assigned prefixes to IA_PDs.

The Hop Limit of packets that contain the DHCPv6 data MUST be 255 to
satisfy the properties of the cellular infrastructure. To reach the
PGW from the UE the DHCPv6 packets are encapsulated by SGW into UDP/
IPv4 packets; this encapsulation is for the GTP-U tunnel. The
corresponding decapsulation mechanism decreases the Hop Limit; when
the Hop Limit reaches value 0 the packet is discarded; to avoid this
situation it is required to put the Hop Limit value of the DHCPv6
Solicit equal to 255.

4.1.3. Option: PPP use

It is possible to use IPv6-over-PPP protocol [RFC1661], with LCP,
between the ARM and the modem. This protocol helps with forming an
IPv6 link-local address on the IoT Router's RI.

```
             ARM       Modem       eNodeB      SGW  PGW
             |         |           |           |    |
             |------->|           |           |    |
             |PPP LCP |           |           |    |
             |<-------|           |           |    |
             |         |           |           |    |
             |         |           |           |    |
             |         |           |           |    |
```

## 4.2. Assignment of the network prefixes on the links

The receipt of the IA_PD Prefix option triggers the GUA
autoconfiguration on the Requesting IoT Router's interfaces. The
Recipient Interface (RI) receives a message with the IA_PD prefix
option and does not perform the autoconfiguration on the current
stage.

All interfaces, except RI, now follow the GUA autoconfiguration
procedure. The number of interfaces that should follow the procedure
could be specified in the configuration file of the Requesting IoT
Router.

The GUA interface autoconfiguration procedure in the Requesting IoT
Router is done by assigning the network addresses from different
GUNPs to the links. The assignment of network addresses is performed
using the 2001:DB8:XXXX:XX::/56 network pool. Therefore, the
Requesting IoT Router operates on multiple links (ingress links).

The IoT Router derives several GUNPs from the received pool. For
example, from the pool 2001:db8:XYZU:TVWZ::/56 the GUNPs
2001:db8:XYZU:TV01::/64 and 2001:db8:XYZU:TV02::/64 are derived.

Further, the GUNPs are aimed at links. For example, the GUNP
2001:DB8:XXXX:XX01::/64 is aimed at the interface 1, and
2001:DB8:XXXX:XX02::/64 at the interface 2; further,
2001:DB8:XXXX:XXff::/64 corresponds to the interface interface 256.

```
          ++======================++
           II                     II
           II                     II
           II                     II
           II                     II                      |H
           II        Prefix 1    +-----+       Link 1      |o
           II         +-------+   | I1 |.....................|s
           II         |  /64  |   +-----+  |                |t
           II         |       |     II    |2001:DB8:XXXX:XX01::/64  |
           II         |       +--------------+              |1
           II         |             II
           II         |             II
           II         |             II
           II         |             II                      |H
    /56  +-----+     |Prefix 2    +-----+       Link 2      |o
   ......|  I0 |----+-------+   | I2 |.....................|s
       +-----+    |  /64  |   +-----+  |                |t
           II        |       |     II    |2001:DB8:XXXX:XX02::/64  |
           II        |       +--------------+              |2
           II        |             II
           II        |             II
           II        |             II
           II        |             II                      |H
           II        |Prefix 3    +-----+       Link 3      |o
           II         +-------+   | I3 |.....................|s
           II           /64  |   +-----+  |                |t
           II               |     II    |2001:DB8:XXXX:XX::/64   |
           II               +--------------+              |3
           II                     II
           II                     II
          ++======================++
              Requesting IoT Router
```
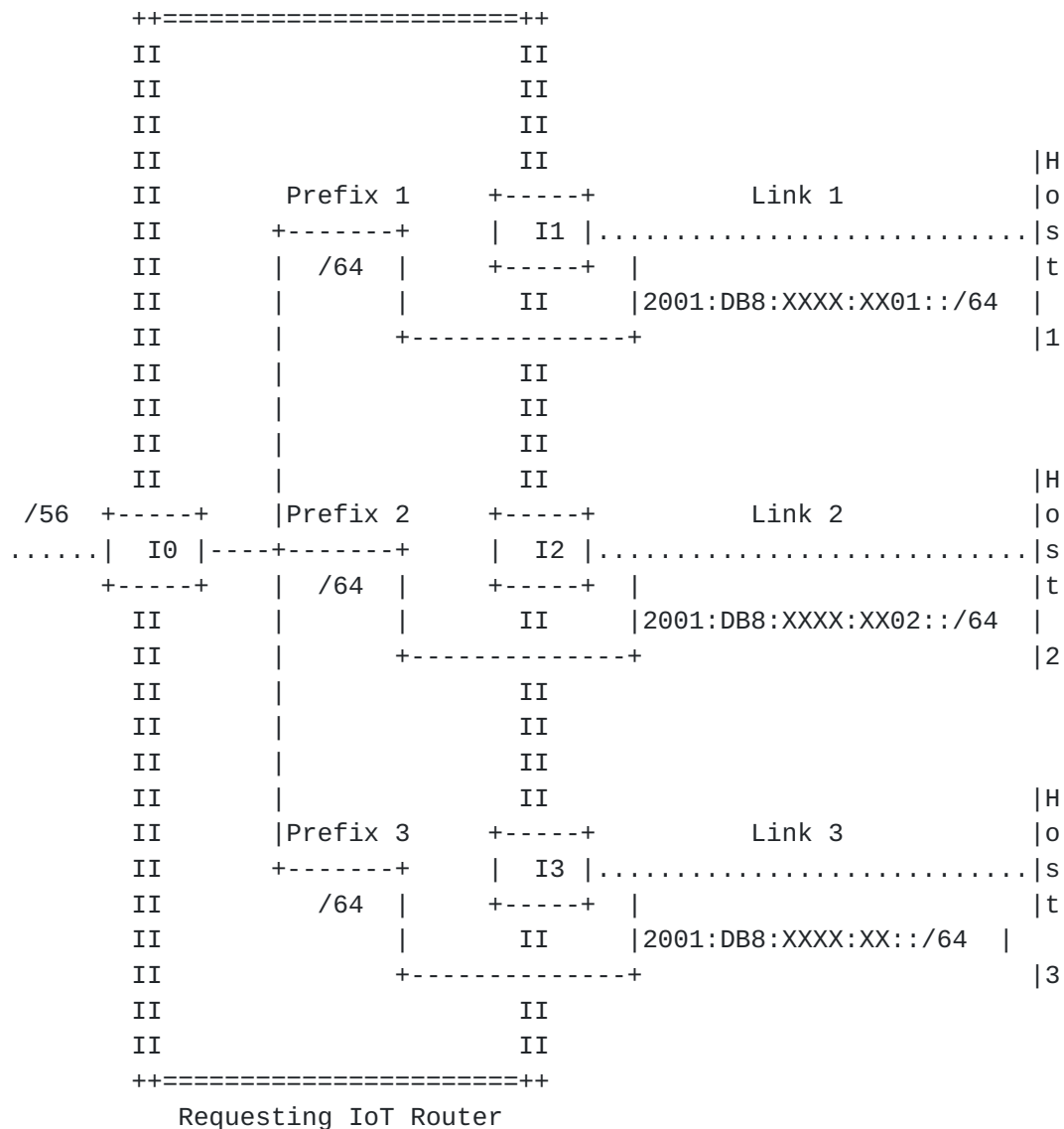
(In the above figure the Requesting IoT Router, with 4 links, is
presented).

There are 2 cases that could be considered. The first one is: all
interfaces (except the RI) of the Requesting IoT Router could be
assigned with the corresponding GUNPs. In the second case, a list of
manually selected interfaces is obtained. Finally interfaces in the
list are assigned with GUNPs. The interfaces numbers represent their
position in alphabetically ordered list by the names that operating
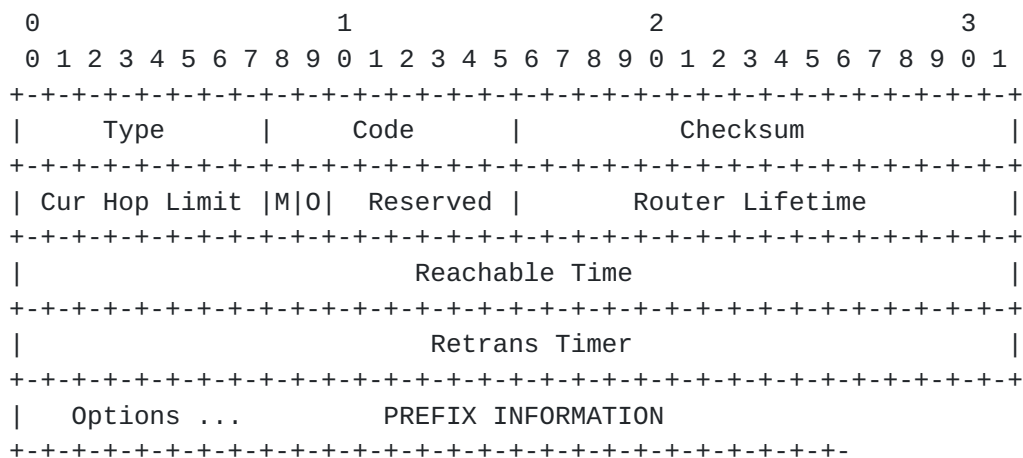system assigned to them during the initialization.

During the final step of the GUA autoconfiguration in the Requesting
IoT Router, the assignment of the IPv6 network addresses to the
interfaces is performed. The IPv6 network addresses have the current

format: 2001:DB8::XXXX:XXYY:1/56, where YY - is a value that
represents the number of the interface. When the GUA
autoconfiguration on the Requesting IoT Router is finished, the
Requesting IoT Router advertises via its interfaces the GUNPs to the
Hosts.

4.3. Advertisement of the network prefixes

Finally, the Requesting IoT Router runs the Neighbour Discovery
Protocol. The Neighbour Discovery Protocol messages allow to
advertise the configuration to the hosts. To send the configuration
parameters from the Requesting IoT Router to the hosts, the "Router
Advertisement" type messages are used [RFC4861]. Usually the "Router
Advertisement" messages are triggered by the "Router Solicit"
messages sent from the Hosts to the Requesting IoT Router [RFC4861].

The "Router Advertisement" message includes the "Prefix Information"
option. It is located in the "Options part" of the "Router
Advertisement" message. The position of the "Prefix Information"
option is presented in the figure below.

```
  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      |     Code      |          Checksum             |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | Cur Hop Limit |M|O|  Reserved |       Router Lifetime         |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                        Reachable Time                         |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                        Retrans Timer                          |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |   Options ...        PREFIX INFORMATION
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

(Figure above presents the Router Advertisement message format
[RFC4861])

The "Prefix Information" option carries the GUNP value and length.
These configuration parameters provide on-link GUNPs, used for SLAAC
auto configuration. The Prefix Length is a number that describes the
number of bits which are used to identify the GUNP. And each GUNP
represents the sub-network.

```
     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      | Prefix Length |L|A| Reserved1 |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                         Valid Lifetime                         |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                       Preferred Lifetime                       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                          Reserved2                             |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                                |
    +                                                                +
    |                                                                |
    +                            Prefix                              +
    |                                                                |
    +                                                                +
    |                                                                |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

4.4. DHCPv6 Port range

   Due to the blockage of the standard port in particular cases there is
   a need to provide a solution that may overcome this issue.

4.4.1. Client support

   On the client side the port range supported in the modified sowtware.
   The DHCPv6 client MAY have the list of dhcpv6 server ports. Client
   send the "SOLICIT" message to the specific port and after [timeout |
   number of retries] the port is changed to the next one in the list.

4.4.2. Server support

   On the server side the port rage support is provided with multiple
   standard DHCPv6 server instances that are running in linux containers
   and listen on different ports.

4.5. Recommendations

   First recommendation we suggest for DHCP is to use port different
   from 547 because it is blocked in many cases. Insted we propose to
   use another one, like 25474. Different approach to address this
   problem is to develop a method to dynamically negotiate such port
   number.

Second recommendation may have to do with Hop Limit and multicast scope in DHCP Solicit messages. It is reasonable to set the value of Hop Limit to 16. It will allow to put DHCP servers _outside_ the cellular network domain.

Third recommendation is that link layer protocol used to make IPv6 addresses in 3GPP network should be documented - and agreed - at IETF, not only at 3GPP. Thus document will be a base for disscussions about IPv6 and User Equipments (UE).

5. Security Considerations

At this time, no security considerations are addressed by this memo.

6. IANA Considerations

No request to IANA at this time.

7. Acknowledgements

The authors would like to thank (in alphabetical order) Michael Mathias Boc, Giorgio Campo, David Frey and Artiol Kalca for their valuable comments related to the Linux Network stack, the Legato OS recipies and the cross-compilation for the ARM architecture. Also the authors would like to acknowledge the contribution of Fred Baker, Michele Russo and Ole Troan for valuable comments during the review of this memo.

8. Normative References

[ETSI102361]
          "ETSI TS 102 361-3 v1.1.7 (2007-12): Electromagnetic
          compatibility and Radio spectrum Matters; Digital Mobile
          Radio Systems; DRM data protocol.".

[I-D.ietf-dhc-rfc3315bis]
          Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A.,
          Richardson, M., Jiang, S., Lemon, T., and T. Winters,
          "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)
          bis", draft-ietf-dhc-rfc3315bis-13 (work in progress),
          April 2018.

[RFC1661] Simpson, W., Ed., "The Point-to-Point Protocol (PPP)",
          STD 51, RFC 1661, DOI 10.17487/RFC1661, July 1994,
          <https://www.rfc-editor.org/info/rfc1661>.

   [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>.

   [RFC3633]   Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic
               Host Configuration Protocol (DHCP) version 6", RFC 3633,
               DOI 10.17487/RFC3633, December 2003,
               <https://www.rfc-editor.org/info/rfc3633>.

   [RFC4861]   Narten, T., Nordmark, E., Simpson, W., and H. Soliman,
               "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861,
               DOI 10.17487/RFC4861, September 2007,
               <https://www.rfc-editor.org/info/rfc4861>.

   [RFC4862]   Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless
               Address Autoconfiguration", RFC 4862,
               DOI 10.17487/RFC4862, September 2007,
               <https://www.rfc-editor.org/info/rfc4862>.

   [RFC6459]   Korhonen, J., Ed., Soininen, J., Patil, B., Savolainen,
               T., Bajko, G., and K. Iisakkila, "IPv6 in 3rd Generation
               Partnership Project (3GPP) Evolved Packet System (EPS)",
               RFC 6459, DOI 10.17487/RFC6459, January 2012,
               <https://www.rfc-editor.org/info/rfc6459>.

   [RFC6653]   Sarikaya, B., Xia, F., and T. Lemon, "DHCPv6 Prefix
               Delegation in Long-Term Evolution (LTE) Networks",
               RFC 6653, DOI 10.17487/RFC6653, July 2012,
               <https://www.rfc-editor.org/info/rfc6653>.

   [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>.

Authors' Addresses

   Dmytro Shytyi
   SFR
   Paris area , Ile-de-France
   France

   Email: ietf.dmytro@shytyi.net
   URI: http://dmytro.shytyi.net

Alexandre Petrescu
CEA, LIST
CEA Saclay
Gif-sur-Yvette , Ile-de-France   91190
France

Phone: +33169089223
Email: Alexandre.Petrescu@cea.fr