Internet Engineering Task Force (IETF)
Request for Comments: 8885
Category: Experimental
ISSN: 2070-1721

CJ. Bernardos
A. de la Oliva
UC3M
F. Giust
Athonet
JC. Ziga
SIGFOX
A. Mourad
InterDigital

October 2020

Proxy Mobile IPv6 Extensions for Distributed Mobility Management

Abstract

Distributed Mobility Management solutions allow networks to be set up in such a way that traffic is distributed optimally and centrally deployed anchors are not relied upon to provide IP mobility support.

There are many different approaches to address Distributed Mobility Management -- for example, extending network-based mobility protocols (like Proxy Mobile IPv6) or client-based mobility protocols (like Mobile IPv6), among others. This document follows the former approach and proposes a solution based on Proxy Mobile IPv6, in which mobility sessions are anchored at the last IP hop router (called the mobility anchor and access router). The mobility anchor and access router is an enhanced access router that is also able to operate as a local mobility anchor or mobility access gateway on a per-prefix basis. The document focuses on the required extensions to effectively support the simultaneous anchoring several flows at different distributed gateways.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for examination, experimental implementation, and evaluation.

This document defines an Experimental Protocol for the Internet community. This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are candidates for any level of Internet Standard; see Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8885.

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

The Distributed Mobility Management (DMM) paradigm aims at minimizing the impact of currently standardized mobility management solutions, which are centralized (at least to a considerable extent) [RFC7333].

The two most relevant examples of current IP mobility solutions are Mobile IPv6 [RFC6275] and Proxy Mobile IPv6 (PMIPv6) [RFC5213]. These solutions offer mobility support at the cost of handling operations at a cardinal point (i.e., the mobility anchor) and burdening it with data forwarding and control mechanisms for a large number of users. The mobility anchor is the home agent for Mobile IPv6 and the local mobility anchor for PMIPv6. As stated in [RFC7333], centralized mobility solutions are prone to several problems and limitations: longer (sub-optimal) routing paths, scalability problems, signaling overhead (and most likely a longer associated handover latency), more complex network deployment, higher vulnerability due to the existence of a potential single point of failure, and lack of granularity of the mobility management service (i.e., mobility is offered on a per-node basis because it is not possible to define finer granularity policies, for example, on a per-application basis).

The purpose of DMM is to overcome the limitations of the traditional centralized mobility management [RFC7333] [RFC7429]; the main concept behind DMM solutions is indeed bringing the mobility anchor closer to the mobile node (MN). Following this idea, the central anchor is moved to the edge of the network and is deployed in the default gateway of the MN. That is, the first elements that provide IP connectivity to a set of MNs are also the mobility managers for those MNs. In this document, we call these entities Mobility Anchors and Access Routers (MAARs).

This document focuses on network-based DMM; hence, the starting point is making PMIPv6 work in a distributed manner [RFC7429]. Mobility is handled by the network without the MN's involvement. But differently from PMIPv6, when the MN moves from one access network to another, the router anchoring the MN's address may change, hence requiring signaling between the anchors to retrieve the MN's previous location(s). Also, a key aspect of network-based DMM is that a prefix pool belongs exclusively to each MAAR in the sense that those prefixes are assigned by the MAAR to the MNs attached to it and are routable at that MAAR. Prefixes are assigned to MNs attached to a MAAR at that time, but remain with those MNs as mobility occurs, remaining always routable at that MAAR as well as towards the MN itself.

We consider partially distributed schemes, where only the data plane is distributed among access routers similar to mobile access gateways (MAGs), whereas the control plane is kept centralized towards a cardinal node (used as an information store), which is discharged from any route management and MN's data forwarding tasks.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Terminology

The following terms used in this document are defined in the PMIPv6 specification [RFC5213]:

BCE: Binding Cache Entry

LMA: Local Mobility Anchor

MAG: Mobile Access Gateway

MN: Mobile Node

P-CoA: Proxy Care-of Address

PBA: Proxy Binding Acknowledgement

PBU: Proxy Binding Update

The following terms used in this document are defined in the Mobile IPv6 (MIPv6) specification [RFC6275]:

CN: Correspondent Node

The following terms are used in this document:

Home Control-Plane Anchor (Home-CPA or H-CPA):
The Home-CPA function hosts the MN's mobility session. There can be more than one mobility session for an MN, and those sessions may be anchored on the same or different Home-CPAs. The Home-CPA will interface with the Home-DPA for managing the forwarding state.

Home Data Plane Anchor (Home-DPA or H-DPA):
The Home-DPA is the topological anchor for the MN's IP addresses and/or prefixes. The Home-DPA is chosen by the Home-CPA on a session basis. The Home-DPA is in the forwarding path for all the

MN's IP traffic.

Access Control Plane Node (Access-CPN or A-CPN):
The Access-CPN is responsible for interfacing with the MN's Home-CPA and the Access-DPN. The Access-CPN has a protocol interface to the Home-CPA.

Access Data Plane Node (Access-DPN or A-DPN):
The Access-DPN function is hosted on the first-hop router where the MN is attached. This function is not hosted on a Layer 2 (L2) bridging device such as an eNode(B) or Access Point.

The following terms are defined and used in this document:

MAAR (Mobility Anchor and Access Router):
First-hop router where the MNs attach. It also plays the role of mobility manager for the IPv6 prefixes it anchors, running the functionalities of PMIP's MAG and LMA. Depending on the prefix, it plays the role of Access-DPN, Home-DPA, and Access-CPN.

CMD (Central Mobility Database):
The node that stores the BCEs allocated for the MNs in the mobility domain. It plays the role of Home-CPA.

P-MAAR (Previous MAAR):
When an MN moves to a new point of attachment, a new MAAR might be allocated as its anchor point for future IPv6 prefixes. The MAAR that served the MN prior to new attachment becomes the P-MAAR. It is still the anchor point for the IPv6 prefixes it had allocated to the MN in the past and serves as the Home-DPA for flows using these prefixes. There might be several P-MAARs serving an MN in cases when the MN is frequently switching points of attachment while maintaining long-lasting flows.

S-MAAR (Serving MAAR):
The MAAR to which the MN is currently attached. Depending on the prefix, it plays the role of Access-DPN, Home-DPA, and Access-CPN.

Anchoring MAAR:
A MAAR anchoring an IPv6 prefix used by an MN.

DLIF (Distributed Logical Interface):
It is a logical interface at the IP stack of the MAAR. For each active prefix used by the MN, the S-MAAR has a DLIF configured (associated with each MAAR still anchoring flows). In this way, an S-MAAR exposes itself towards each MN as multiple routers, one as itself and one per P-MAAR.

3. PMIPv6 DMM Extensions

The solution consists of decoupling the entities that participate in the data and the control planes: the data plane becomes distributed and managed by the MAARs near the edge of the network, while the control plane, besides those on the MAARs, relies on a central entity called the Central Mobility Database (CMD). In the proposed architecture, the hierarchy present in PMIPv6 between LMA and MAG is preserved but with the following substantial variations:

* The LMA is discharged from the data forwarding role; only the Binding Cache and its management operations are maintained. Hence, the LMA is renamed as "CMD", which is therefore a Home-CPA. Also, the CMD is able to send and parse both PBU and PBA messages.

* The MAG is enriched with the LMA functionalities, hence the name Mobility Anchor and Access Router (MAAR). It maintains a local Binding Cache for the MNs that are attached to it, and it is able

to send and parse PBU and PBA messages.

* The Binding Cache will be extended to include information regarding P-MAARs where the MN was anchored and still retains active data sessions.

* Each MAAR has a unique set of global prefixes (which are configurable) that can be allocated by the MAAR to the MNs but must be exclusive to that MAAR, i.e., no other MAAR can allocate the same prefixes.

The MAARs leverage the CMD to access and update information related to the MNs, which is stored as mobility sessions; hence, a centralized node maintains a global view of the network status. The CMD is queried whenever an MN is detected joining/leaving the mobility domain. It might be a fresh attachment, a detachment, or a handover, but as MAARs are not aware of past information related to a mobility session, they contact the CMD to retrieve the data of interest and eventually take the appropriate action. The procedure adopted for the query and the message exchange sequence might vary to optimize the update latency and/or the signaling overhead. Here, one method for the initial registration and three different approaches for updating the mobility sessions using PBUs and PBAs are presented. Each approach assigns a different role to the CMD:

* The CMD is a PBU/PBA relay;

* The CMD is only a MAAR locator;

* The CMD is a PBU/PBA proxy.

The solution described in this document allows per-prefix anchoring decisions -- for example, to support the anchoring of some flows at a central Home-DPA (like a traditional LMA) or to enable an application to switch to the locally anchored prefix to gain route optimization, as indicated in [RFC8563]. This type of per-prefix treatment would potentially require additional extensions to the MAARs and signaling between the MAARs and the MNs to convey the per-flow anchor preference (central, distributed), which are not covered in this document.

Note that an MN may move across different MAARs, which might result in several P-MAARs existing at a given moment of time, each of them anchoring a different prefix used by the MN.

3.1. Initial Registration

Initial registration is performed when an MN attaches to a network for the first time (rather than attaching to a new network after moving from a previous one).

In this description (shown in Figure 1), it is assumed that:

1. The MN is attaching to MAAR1.

2. The MN is authorized to attach to the network.

Upon MN attachment, the following operations take place:

1. MAAR1 assigns a global IPv6 prefix from its own prefix pool to the MN (Pref1). It also stores this prefix (Pref1) in the locally allocated temporary BCE.

2. MAAR1 sends a PBU [RFC5213] with Pref1 and the MN’s MN-ID to the CMD.

3. Since this is an initial registration, the CMD stores a BCE containing the MN-ID, Pref1, and MAAR1's address (as a Proxy-CoA) as the primary fields.

4. The CMD replies with a PBA with the usual options defined in PMIPv6 [RFC5213], meaning that the MN's registration is fresh and no past status is available.

5. MAAR1 stores the BCE described in (1) and unicasts a Router Advertisement (RA) to the MN with Pref1.

6. The MN uses Pref1 to configure an IPv6 address (IP1) (e.g., with stateless address autoconfiguration (SLAAC)).

Note that:

1. Alternative IPv6 autoconfiguration mechanisms can also be used, though this document describes the SLAAC-based one.

2. IP1 is routable at MAAR1 in the sense that it is on the path of packets addressed to the MN.

3. MAAR1 acts as a plain router for packets destined to the MN as no encapsulation or special handling takes place.

In the diagram shown in Figure 1 (and subsequent diagrams), the flow of packets is presented using '*'.

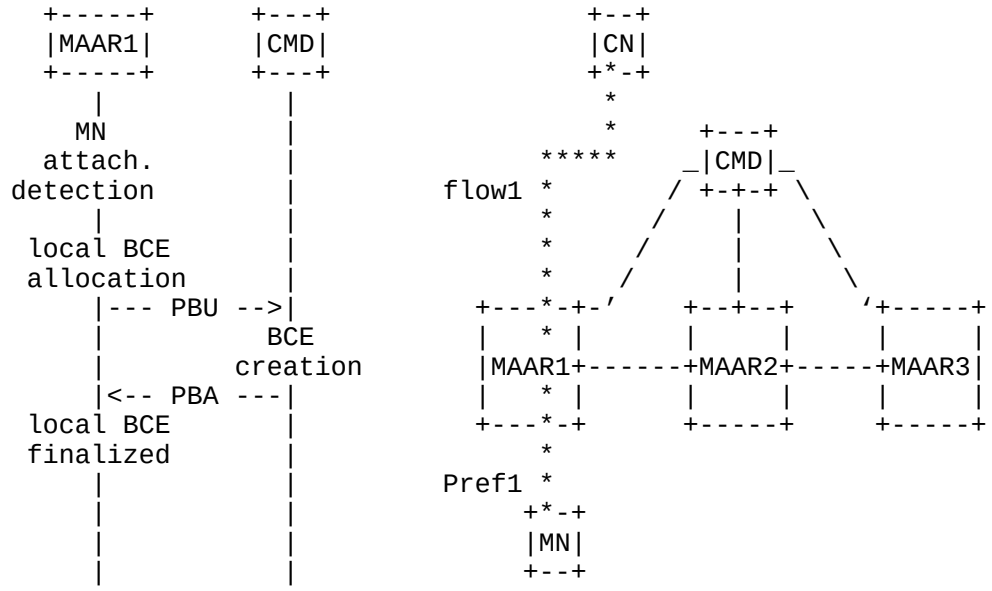


Figure 1: First Attachment to the Network

Note that the registration process does not change regardless of the CMD's modes (relay, locator, or proxy) described in the following sections. The procedure is depicted in Figure 1.

3.2. The CMD as PBU/PBA Relay

Upon MN mobility, if the CMD behaves as a PBU/PBA relay, the following operations take place:

1. When the MN moves from its current point of attachment and attaches to MAAR2 (now the S-MAAR), MAAR2 reserves an IPv6 prefix (Pref2), stores a temporary BCE, and sends a PBU to the CMD for registration.

2. Upon PBU reception and BC lookup, the CMD retrieves an already existing entry for the MN and binds the MN-ID to its former location; thus, the CMD forwards the PBU to the MAAR indicated as Proxy-CoA (MAAR1) and includes a new mobility option to communicate the S-MAAR's global address to MAAR1 (defined as the Serving MAAR option in Section 4.6). The CMD updates the P-CoA field in the BCE related to the MN with the S-MAAR's address.

3. Upon PBU reception, MAAR1 can install a tunnel on its side towards MAAR2 and the related routes for Pref1. Then MAAR1 replies to the CMD with a PBA (including the option mentioned before) to ensure that the new location has successfully changed. The PBA contains the prefix anchored at MAAR1 in the Home Network Prefix option.

4. The CMD, after receiving the PBA, updates the BCE and populates an instance of the P-MAAR list. The P-MAAR list is an additional field on the BCE that contains an element for each P-MAAR involved in the MN's mobility session. The list element contains the P-MAAR's global address and the prefix it has delegated. Also, the CMD sends a PBA to the new S-MAAR, which contains the previous Proxy-CoA and the prefix anchored to it embedded into a new mobility option called the Previous MAAR option (defined in Section 4.5). Then, upon PBA arrival, a bidirectional tunnel can be established between the two MAARs, and new routes are set appropriately to recover the IP flow(s) carrying Pref1.

5. Now, packets destined for Pref1 are first received by MAAR1, encapsulated into the tunnel, and forwarded to MAAR2, which finally delivers them to their destination. In the uplink, when the MN transmits packets using Pref1 as a source address, they are sent to MAAR2 (as it is the MN's new default gateway) and then tunneled to MAAR1, which routes them towards the next hop to the destination. Conversely, packets carrying Pref2 are routed by MAAR2 without any special packet handling both for the uplink and downlink.

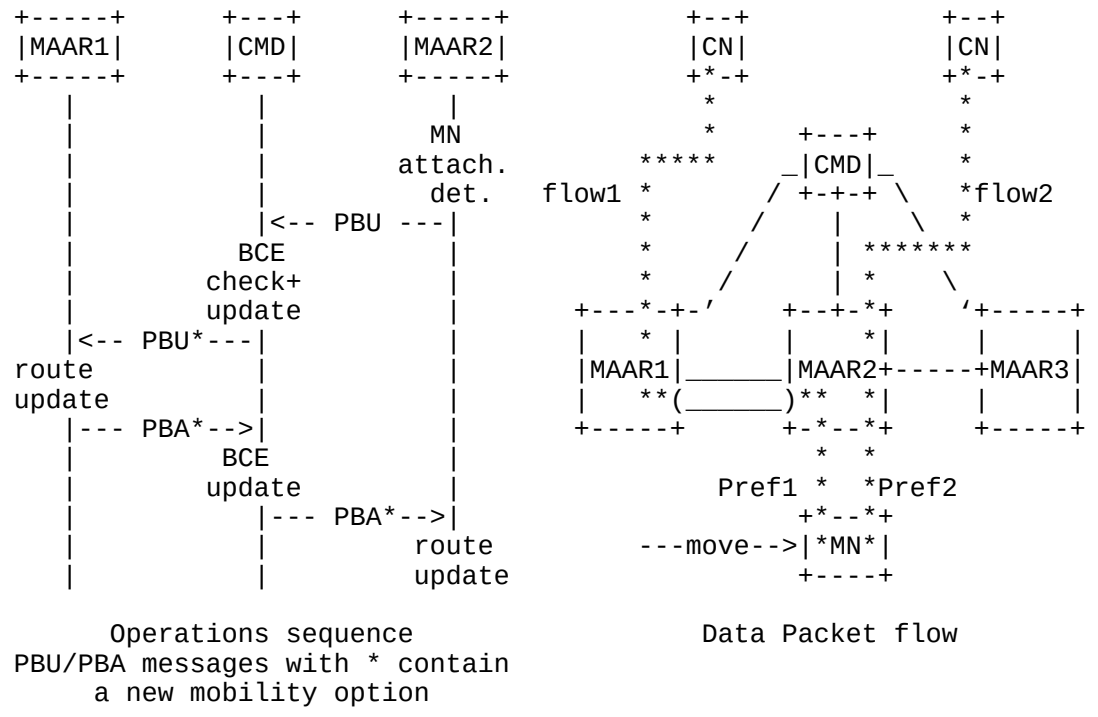


Figure 2: Scenario after a Handover, CMD as Relay

For MN's next movements, the process is repeated, but the number of

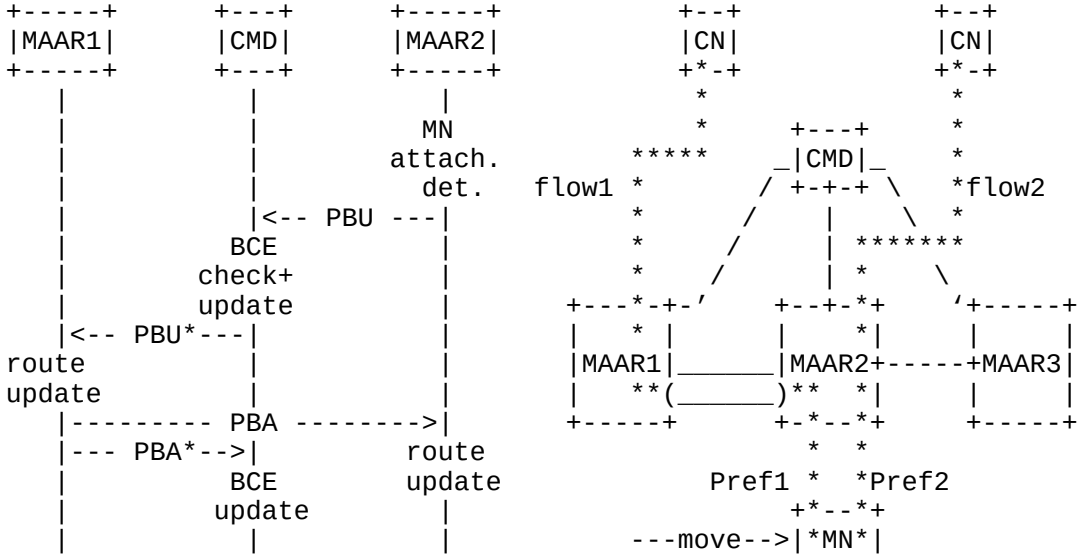
P-MAARs involved increases (according to the number of prefixes that the MN wishes to maintain). Indeed, once the CMD receives the first PBU from the new S-MAAR, it forwards copies of the PBU to all the P-MAARs indicated in the BCE, namely the P-MAAR registered as the current P-CoA (i.e., the MAAR prior to handover) plus the ones in the P-MAAR list. Those P-MAARs reply with a PBA to the CMD, which aggregates all of the PBAs into one PBA to notify the S-MAAR, which finally can establish the tunnels with the P-MAARs.

It should be noted that this design separates the mobility management at the prefix granularity, and it can be tuned in order to erase old mobility sessions when not required, while the MN is reachable through the latest prefix acquired. Moreover, the latency associated with the mobility update is bound to the PBA sent by the furthest P-MAAR, in terms of RTT, that takes the longest time to reach the CMD. The drawback can be mitigated by introducing a timeout at the CMD, by which, after its expiration, all the PBAs so far collected are transmitted, and the remaining are sent later upon their arrival. Note that, in this case, the S-MAAR might receive multiple PBAs from the CMD in response to a PBU. The CMD SHOULD follow the retransmissions and rate-limiting considerations described in Section 3.6, especially when aggregating and relaying PBAs.

When there are multiple P-MAARs, e.g., k MAARs, a single PBU received by the CMD triggers k outgoing packets from a single incoming packet. This may lead to packet bursts originating from the CMD, albeit to different targets. Pacing mechanisms MUST be introduced to avoid bursts on the outgoing link.

3.3. The CMD as MAAR Locator

The handover latency experienced in the approach shown before can be reduced if the P-MAARs are allowed to directly signal their information to the new S-MAAR. This procedure reflects what was described in Section 3.2 up to the moment the P-MAAR receives the PBU with the Serving MAAR option. At that point, a P-MAAR is aware of the new MN's location (because of the S-MAAR's address in the Serving MAAR option), and, besides sending a PBA to the CMD, it also sends a PBA to the S-MAAR, including the prefix it is anchoring. This latter PBA does not need to include new options, as the prefix is embedded in the Home Network Prefix (HNP) option and the P-MAAR's address is taken from the message's source address. The CMD is released from forwarding the PBA to the S-MAAR as the latter receives a copy directly from the P-MAAR with the necessary information to build the tunnels and set the appropriate routes. Figure 3 illustrates the new message sequence. The data forwarding is unaltered.



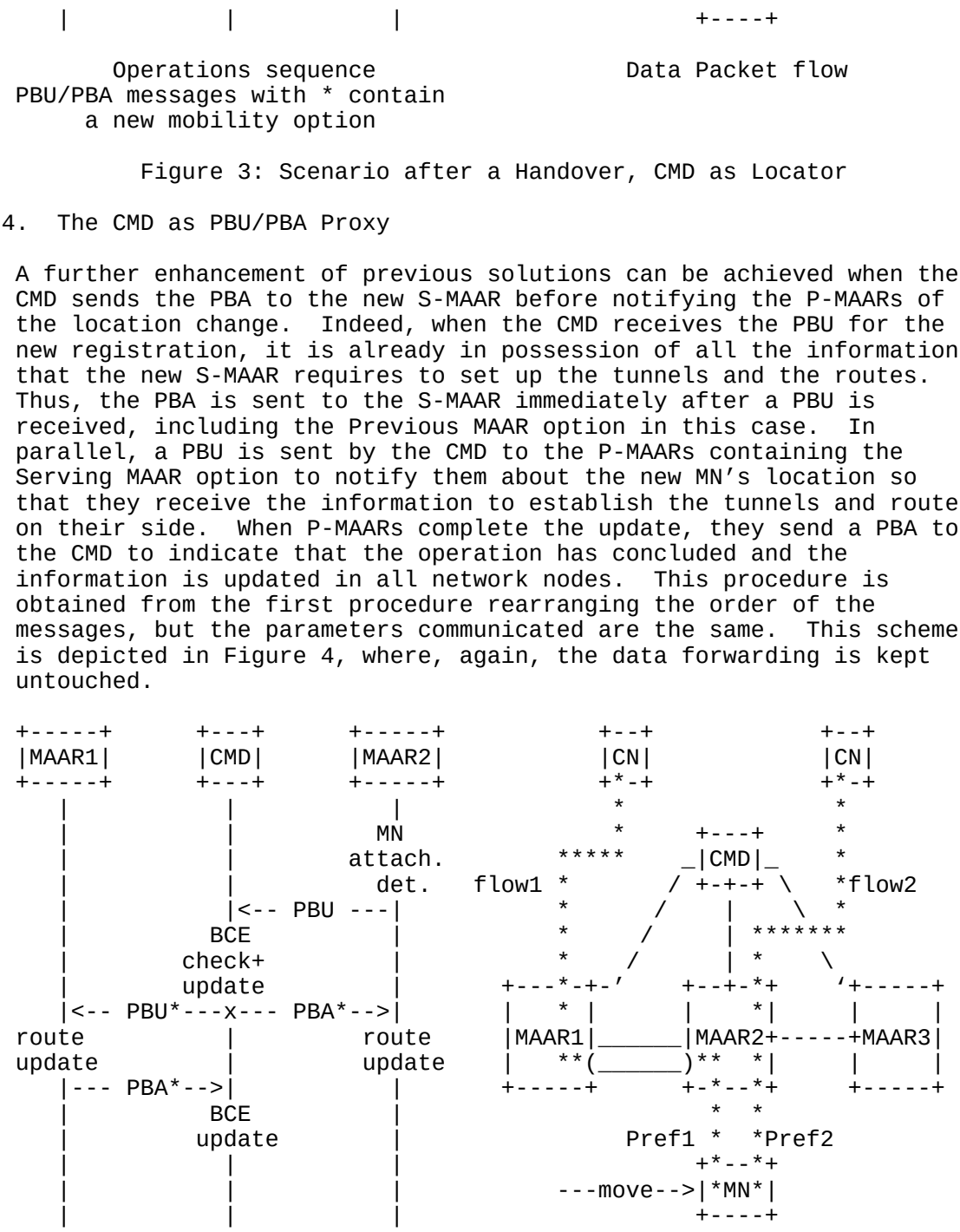


Figure 3: Scenario after a Handover, CMD as Locator

3.4. The CMD as PBU/PBA Proxy

A further enhancement of previous solutions can be achieved when the CMD sends the PBA to the new S-MAAR before notifying the P-MAARs of the location change. Indeed, when the CMD receives the PBU for the new registration, it is already in possession of all the information that the new S-MAAR requires to set up the tunnels and the routes. Thus, the PBA is sent to the S-MAAR immediately after a PBU is received, including the Previous MAAR option in this case. In parallel, a PBU is sent by the CMD to the P-MAARs containing the Serving MAAR option to notify them about the new MN's location so that they receive the information to establish the tunnels and routes on their side. When P-MAARs complete the update, they send a PBA to the CMD to indicate that the operation has concluded and the information is updated in all network nodes. This procedure is obtained from the first procedure rearranging the order of the messages, but the parameters communicated are the same. This scheme is depicted in Figure 4, where, again, the data forwarding is kept untouched.

Figure 4: Scenario after a Handover, CMD as Proxy

3.5. De-registration

The de-registration mechanism devised for PMIPv6 cannot be used as is in this solution because each MAAR handles an independent mobility session (i.e., a single prefix or a set of prefixes) for a given MN, whereas the aggregated session is stored at the CMD. Indeed, if a P-MAAR initiates a de-registration procedure because the MN is no longer present on the MAAR's access link, it removes the routing state for the prefix(es), that would be deleted by the CMD as well, hence defeating any prefix continuity attempt. The simplest approach to overcome this limitation is to deny a P-MAAR to de-register a

prefix, that is, allowing only an S-MAAR to de-register the whole MN session. This can be achieved by first removing any L2 detachment event so that de-registration is triggered only when the binding lifetime expires, hence providing a guard interval for the MN to connect to a new MAAR. Then, a change in the MAAR operations is required, and at this stage, two possible solutions can be deployed:

* A P-MAAR stops the BCE timer upon receiving a PBU from the CMD containing a "Serving MAAR" option. In this way, only the S-MAAR is allowed to de-register the mobility session, arguing that the MN definitely left the domain.

* P-MAARs can, upon BCE expiry, send de-registration messages to the CMD, which, instead of acknowledging the message with a 0 lifetime, sends back a PBA with a non-zero lifetime, hence renewing the session if the MN is still connected to the domain.

3.6. Retransmissions and Rate Limiting

The node sending PBUs (the CMD or S-MAAR) SHOULD make use of the timeout to also deal with missing PBAs (to retransmit PBUs). The INITIAL_BINDACK_TIMEOUT [RFC6275] SHOULD be used for configuring the retransmission timer. The retransmissions by the node MUST use an exponential backoff process in which the timeout period is doubled upon each retransmission until either the node receives a response or the timeout period reaches the value MAX_BINDACK_TIMEOUT [RFC6275]. The node MAY continue to send these messages at this slower rate indefinitely. The node MUST NOT send PBU messages to a particular node more than MAX_UPDATE_RATE times within a second [RFC6275].

3.7. The Distributed Logical Interface (DLIF) Concept

One of the main challenges of a network-based DMM solution is how to allow a MN to simultaneously send/receive traffic that is anchored at different MAARs and how to influence the MN's selection process of its source IPv6 address for a new flow without requiring special support from the MN's IP stack. This document defines the DLIF, which is a software construct in the MAAR that can easily hide the change of associated anchors from the MN.

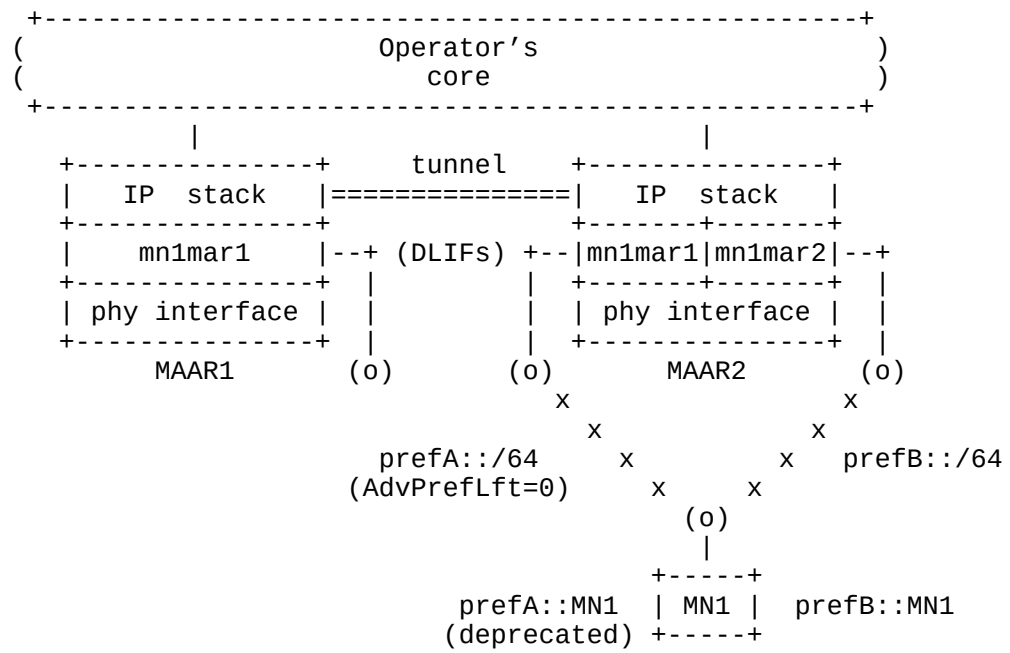


Figure 5: DLIF: Exposing Multiple Routers (One per P-MAAR)

The basic idea of the DLIF concept is the following: each S-MAAR

exposes itself to a given MN as multiple routers, one per P-MAAR associated with the MN. Let's consider the example shown in Figure 5: MN1 initially attaches to MAAR1, configuring an IPv6 address (prefA::MN1) from a prefix locally anchored at MAAR1 (prefA::/64). At this stage, MAAR1 plays the role of both anchoring and serving MAAR and also behaves as a plain IPv6 access router. MAAR1 creates a DLIF to communicate (through a point-to-point link) with MN1, exposing itself as a (logical) router with specific MAC and IPv6 addresses (e.g., prefA::MAAR1/64 and fe80::MAAR1/64) using the DLIF mn1mar1. As explained below, these addresses represent the "logical" identity of MAAR1 for MN1 and will "follow" the MN while roaming within the domain (note that the place where all this information is maintained and updated is out of scope of this document; potential examples are to keep it on the home subscriber server -- HSS -- or the user's profile).

If MN1 moves and attaches to a different MAAR of the domain (MAAR2 in the example of Figure 5), this MAAR will create a new logical interface (mn1mar2) to expose itself to MN1, providing it with a locally anchored prefix (prefB::/64). In this case, since the MN1 has another active IPv6 address anchored at MAAR1, MAAR2 also needs to create an additional logical interface configured to resemble the one used by MAAR1 to communicate with MN1. In this example, MAAR1 is the only P-MAAR (MAAR2 is the same as S-MAAR), so only the logical interface mn1mar1 is created. However, the same process would be repeated if more P-MAARs were involved. In order to keep the prefix anchored at MAAR1 reachable, a tunnel between MAAR1 and MAAR2 is established and the routing is modified accordingly. The PBU/PBA signaling is used to set up the bidirectional tunnel between MAAR1 and MAAR2, and it might also be used to convey the information about the prefix(es) anchored at MAAR1 and the addresses of the associated DLIF (i.e., mn1mar1) to MAAR2.

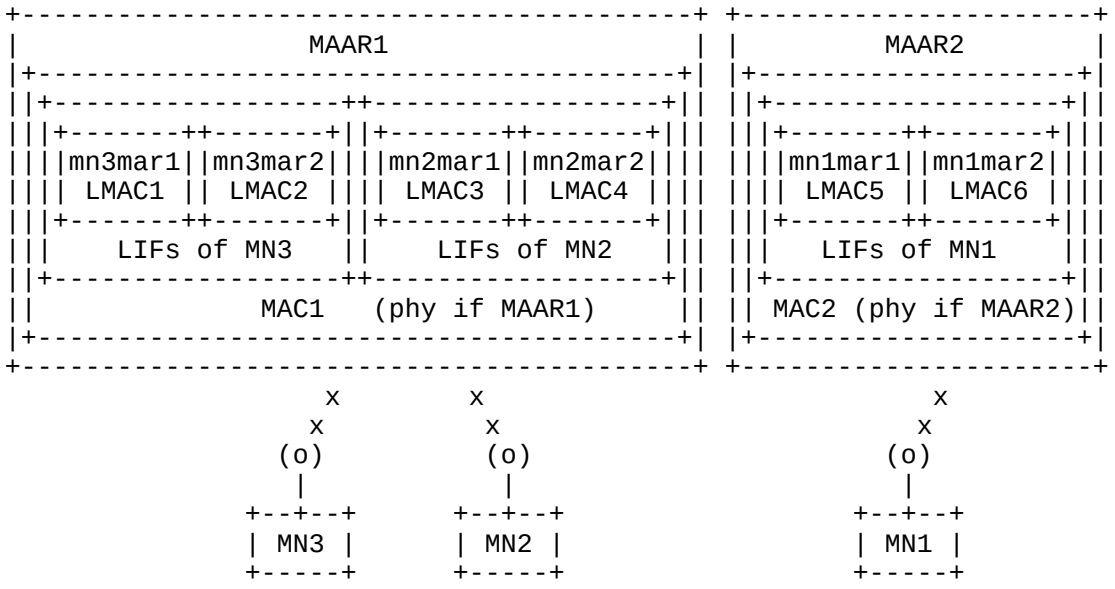


Figure 6: Distributed Logical Interface Concept

Figure 6 shows the logical interface concept in more detail. The figure shows two MAARs and three MNs. MAAR1 is currently serving MN2 and MN3, while MAAR2 is serving MN1. Note that an S-MAAR always plays the role of anchoring MAAR for the attached (served) MNs. Each MAAR has one single physical wireless interface as depicted in this example.

As discussed before, each MN always "sees" multiple logical routers -- one per anchoring MAAR -- independently of its currently S-MAAR. From the point of view of the MN, these MAARs are portrayed as

different routers, although the MN is physically attached to a single interface. This is achieved by the S-MAAR configuring different logical interfaces. MN1 is currently attached to MAAR2 (i.e., MAAR2 is its S-MAAR) and, therefore, it has configured an IPv6 address from MAAR2's pool (e.g., prefB::/64). MAAR2 has set up a logical interface (mn1mar2) on top of its wireless physical interface (phy if MAAR2), which is used to serve MN1. This interface has a logical MAC address (LMAC6) that is different from the hardware MAC address (MAC2) of the physical interface of MAAR2. Over the mn1mar2 interface, MAAR2 advertises its locally anchored prefix prefB::/64. Before attaching to MAAR2, MN1 was attached to MAAR1 and configured a locally anchored address at that MAAR, which is still being used by MN1 in active communications. MN1 keeps "seeing" an interface connecting to MAAR1 as if it were directly connected to the two MAARs. This is achieved by the S-MAAR (MAAR2) configuring an additional DLIF, mn1mar1, which behaves as the logical interface configured by MAAR1 when MN1 was attached to it. This means that both the MAC and IPv6 addresses configured on this logical interface remain the same regardless of the physical MAAR that is serving the MN. The information required by an S-MAAR to properly configure this logical interfaces can be obtained in different ways: as part of the information conveyed in the PBA, from an external database (e.g., the HSS) or by other means. As shown in the figure, each MAAR may have several logical interfaces associated with each attached MN and always has at least one (since an S-MAAR is also an anchoring MAAR for the attached MN).

In order to enforce the use of the prefix locally anchored at the S-MAAR, the RAs sent over those logical interfaces playing the role of anchoring MAARs (different from the serving one) include a zero preferred prefix lifetime (and a non-zero valid prefix lifetime, so the prefix remains valid while being deprecated). The goal is to deprecate the prefixes delegated by these MAARs (so that they will no longer be serving the MN). Note that ongoing communications may keep on using those addresses even if they are deprecated, so this only affects the establishment of new sessions.

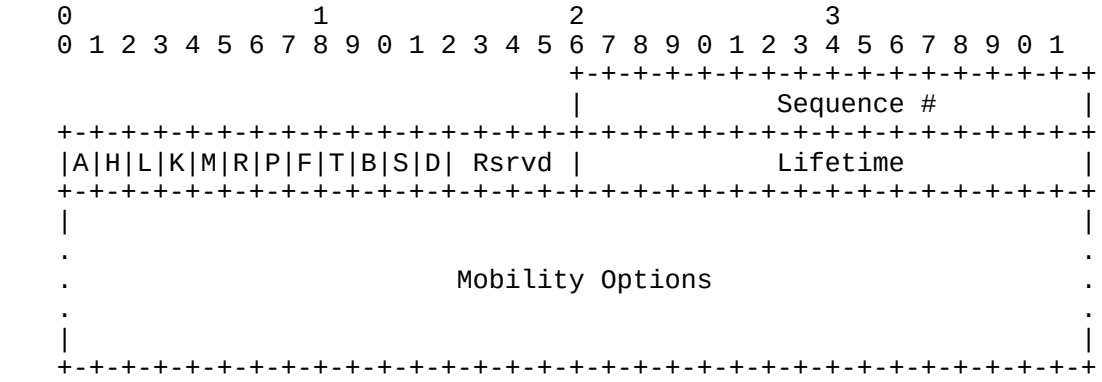
The DLIF concept also enables the following use case: suppose that access to a local IP network is provided by a given MAAR (e.g., MAAR1 in the example shown in Figure 5) and that the resources available at that network cannot be reached from outside the local network (e.g., cannot be accessed by an MN attached to MAAR2). This is similar to the local IP access scenario considered by 3GPP, where a local gateway node is selected for sessions requiring access to services provided locally (instead of going through a central gateway). The goal is to allow an MN to be able to roam while still being able to have connectivity to this local IP network. The solution adopted to support this case makes use of more specific routes, as discussed in RFC 4191 [RFC4191], when the MN moves to a MAAR different from the one providing access to the local IP network (MAAR1 in the example). These routes are advertised through the DLIF where the MAAR is providing access to the local network (MAAR1 in this example). In this way, if MN1 moves from MAAR1 to MAAR2, any active session that MN1 may have with a node on the local network connected to MAAR1 will survive via the tunnel between MAAR1 and MAAR2. Also, any potential future connection attempt to the local network will be supported even though MN1 is no longer attached to MAAR1, so long as a source address configured from MAAR1 is selected for new connections (see [RFC6724], rule 5.5).

4. Message Format

This section defines extensions to the PMIPv6 [RFC5213] protocol messages.

4.1. Proxy Binding Update

A new flag (D) is included in the PBU to indicate that the PBU is coming from a MAAR or a CMD and not from a MAG. The rest of the PBU format remains the same as defined in [RFC5213].

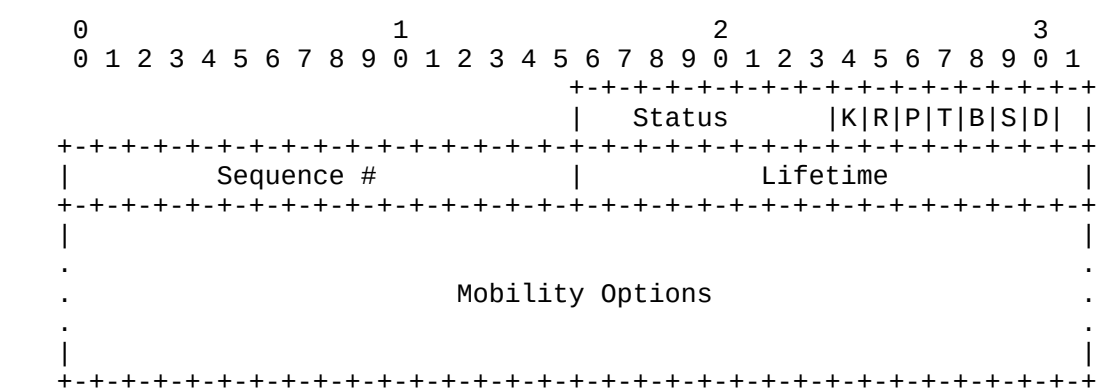


DMM Flag (D)
The D flag is set to indicate to the receiver of the message that the PBU is from a MAAR or a CMD. When an LMA that does not support the extensions described in this document receives a message with the D flag set, the PBU in that case MUST NOT be processed by the LMA, and an error MUST be returned.

Mobility Options
Variable-length field of such length that the complete Mobility Header is an integer that is a multiple of 8 octets long. This field contains zero or more TLV-encoded mobility options. The encoding and format of the defined options are described in Section 6.2 of [RFC6275]. The receiving node MUST ignore and skip any options that it does not understand.

4.2. Proxy Binding Acknowledgement

A new flag (D) is included in the PBA to indicate that the sender supports operating as a MAAR or CMD. The rest of the PBA format remains the same as defined in [RFC5213].



DMM Flag (D)
The D flag is set to indicate that the sender of the message supports operating as a MAAR or CMD. When a MAG that does not support the extensions described in this document receives a message with the D flag set, it MUST ignore the message, and an error MUST be returned.

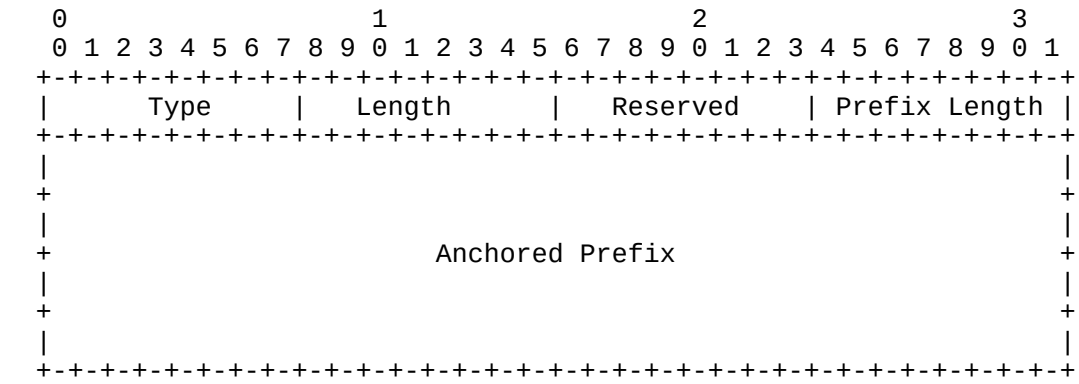
Mobility Options
Variable-length field of such length that the complete Mobility Header is an integer multiple of 8 octets long. This field contains zero or more TLV-encoded mobility options. The encoding and format of the defined options are described in Section 6.2 of

[RFC6275]. The MAAR MUST ignore and skip any options that it does not understand.

4.3. Anchored Prefix Option

A new Anchored Prefix option is defined for use with the PBU and PBA messages exchanged between MAARs and CMDs. Therefore, this option can only appear if the D bit is set in a PBU/PBA. This option is used for exchanging the MN's prefix anchored at the anchoring MAAR. There can be multiple Anchored Prefix options present in the message.

The Anchored Prefix option has an alignment requirement of 8n+4. Its format is as follows:



Type
65

Length
8-bit unsigned integer indicating the length of the option in octets, excluding the type and length fields. This field MUST be set to 18.

Reserved
This field is unused at the time of publication. The value MUST be initialized to 0 by the sender and MUST be ignored by the receiver.

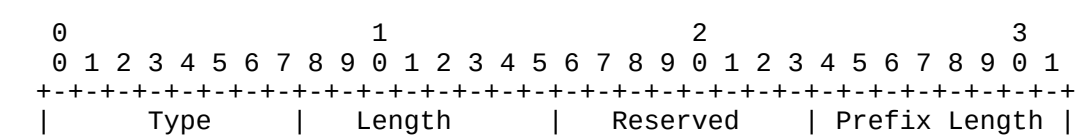
Prefix Length
8-bit unsigned integer indicating the prefix length in bits of the IPv6 prefix contained in the option.

Anchored Prefix
A 16-octet field containing the MN's IPv6 Anchored Prefix. Only the first Prefix Length bits are valid for the Anchored Prefix option. The rest of the bits MUST be ignored.

4.4. Local Prefix Option

A new Local Prefix option is defined for use with the PBU and PBA messages exchanged between MAARs or between a MAAR and a CMD. Therefore, this option can only appear if the D bit is set in a PBU/PBA. This option is used for exchanging a prefix of a local network that is only reachable via the anchoring MAAR. There can be multiple Local Prefix options present in the message.

The Local Prefix option has an alignment requirement of 8n+4. Its format is as follows:



Type
66

Length
8-bit unsigned integer indicating the length of the option in octets, excluding the type and length fields. This field MUST be set to 18.

Reserved
This field is unused at the time of publication. The value MUST be initialized to 0 by the sender and MUST be ignored by the receiver.

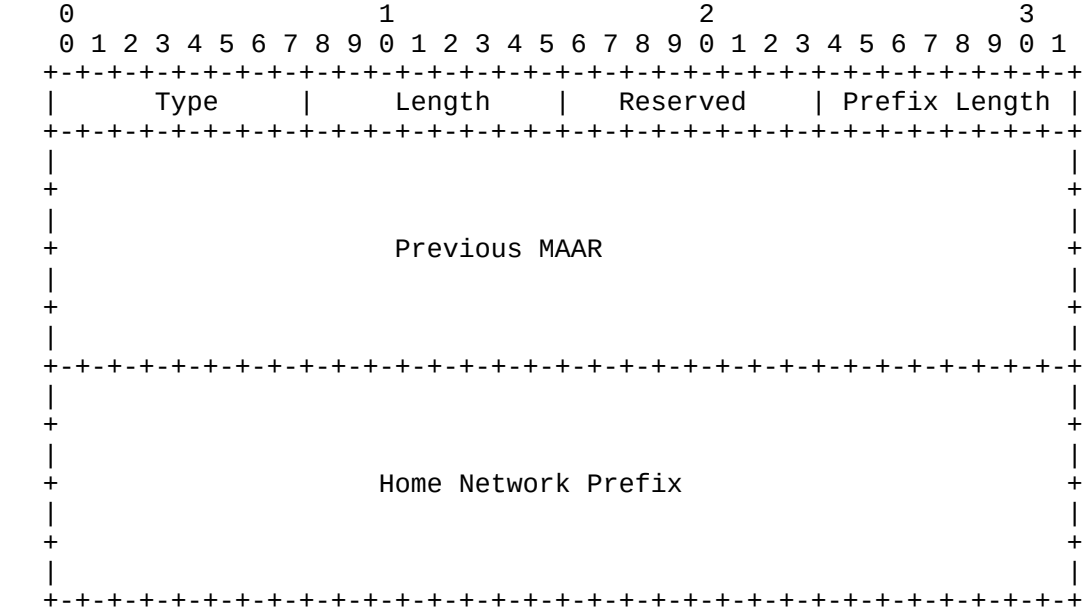
Prefix Length
8-bit unsigned integer indicating the prefix length in bits of the IPv6 prefix contained in the option.

Local Prefix
A 16-octet field containing the IPv6 Local Prefix. Only the first Prefix Length bits are valid for the IPv6 Local Prefix. The rest of the bits MUST be ignored.

4.5. Previous MAAR Option

This new option is defined for use with the PBA messages exchanged by the CMD to a MAAR. This option is used to notify the S-MAAR about the P-MAAR's global address and the prefix anchored to it. There can be multiple Previous MAAR options present in the message.

The Previous MAAR option has an alignment requirement of 8n+4. Its format is as follows:



Type
67

Length
8-bit unsigned integer indicating the length of the option in octets, excluding the type and length fields. This field MUST be set to 34.

Reserved
This field is unused at the time of publication. The value MUST be initialized to 0 by the sender and MUST be ignored by the receiver.

Prefix Length
8-bit unsigned integer indicating the prefix length in bits of the IPv6 prefix contained in the option.

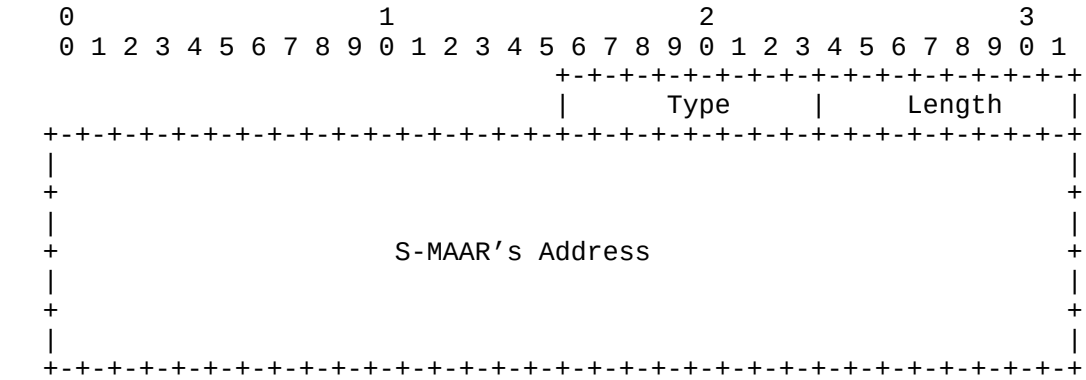
Previous MAAR
A 16-octet field containing the P-MAAR's IPv6 global address.

Home Network Prefix
A 16-octet field containing the MN's IPv6 Home Network Prefix. Only the first Prefix Length bits are valid for the MN's IPv6 Home Network Prefix. The rest of the bits MUST be ignored.

4.6. Serving MAAR Option

This new option is defined for use with the PBU message exchanged between the CMD and a P-MAAR. This option is used to notify the P-MAAR about the current S-MAAR's global address. Its format is as follows:

The Serving MAAR option has an alignment requirement of 8n+6. Its format is as follows:



Type
68

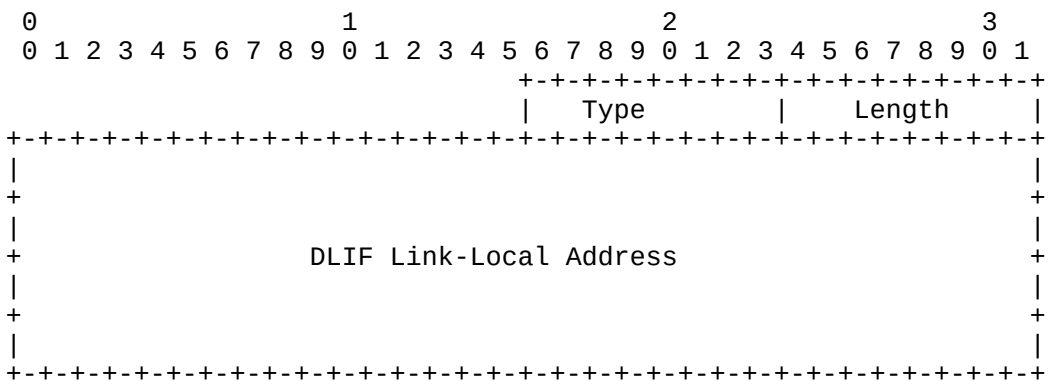
Length
8-bit unsigned integer indicating the length of the option in octets, excluding the type and length fields. This field MUST be set to 16.

Serving MAAR
A 16-octet field containing the S-MAAR's IPv6 global address.

4.7. DLIF Link-Local Address Option

A new DLIF Link-Local Address option is defined for use with the PBA message exchanged between MAARs and between a MAAR and a CMD. This option is used for exchanging the link-local address of the DLIF to be configured on the S-MAAR so it resembles the DLIF configured on the P-MAAR.

The DLIF Link-Local Address option has an alignment requirement of 8n+6. Its format is as follows:



Type
69

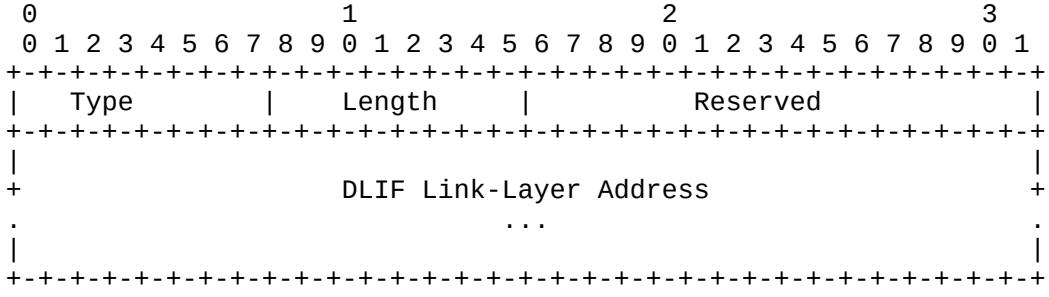
Length
8-bit unsigned integer indicating the length of the option in octets, excluding the type and length fields. This field MUST be set to 16.

DLIF Link-Local Address
A 16-octet field containing the link-local address of the logical interface.

4.8. DLIF Link-Layer Address Option

A new DLIF Link-Layer Address option is defined for use with the PBA message exchanged between MAARs and between a MAAR and a CMD. This option is used for exchanging the link-layer address of the DLIF to be configured on the S-MAAR so it resembles the DLIF configured on the P-MAAR.

The format of the DLIF Link-Layer Address option is shown below. Based on the size of the address, the option MUST be aligned appropriately, as per the mobility option alignment requirements specified in [RFC6275].



Type
70

Length
8-bit unsigned integer indicating the length of the option in octets, excluding the type and length fields.

Reserved
This field is unused at the time of publication. The value MUST be initialized to 0 by the sender and MUST be ignored by the receiver.

DLIF Link-Layer Address

A variable length field containing the link-layer address of the logical interface to be configured on the S-MAAR.

The content and format of this field (including octet and bit ordering) is as specified in Section 4.6 of [RFC4861] for carrying link-layer addresses. On certain access links where the link-layer address is not used or cannot be determined, this option cannot be used.

5. IANA Considerations

This document defines six new mobility options: Anchored Prefix, Local Prefix, Previous MAAR, Serving MAAR, DLIF Link-Local Address, and DLIF Link-Layer Address. IANA has assigned Type values for these options from the same numbering space as allocated for the other mobility options in the "Mobility Options" registry defined in http://www.iana.org/assignments/mobility-parameters.

This document reserves a new flag (D) with a value of 0x0010 in the "Binding Update Flags" registry and a new flag (D) with a value of 0x02 in the "Binding Acknowledgment Flags" of the "Mobile IPv6 parameters" registry (http://www.iana.org/assignments/mobility-parameters).

6. Security Considerations

The protocol extensions defined in this document share the same security concerns of PMIPv6 [RFC5213]. It is recommended that the signaling messages, PBU and PBA, exchanged between the MAARs be protected using IPsec, specifically by using the established security association between them. This essentially eliminates the threats related to the impersonation of a MAAR.

When the CMD acts as a PBU/PBA relay, the CMD may act as a relay of a single PBU to multiple P-MAARs. In situations with many fast handovers (e.g., with vehicular networks), multiple previous (e.g., k) MAARs may exist. In this situation, the CMD creates k outgoing packets from a single incoming packet. This bears a certain amplification risk. The CMD MUST use a pacing approach in the outgoing queue to cap the output traffic (i.e., the rate of PBUs sent) to limit this amplification risk.

When the CMD acts as a MAAR locator, mobility signaling (PBAs) is exchanged between P-MAARs and the current S-MAAR. Hence, security associations are REQUIRED to exist between the involved MAARs (in addition to the ones needed with the CMD).

Since de-registration is performed by timeout, measures SHOULD be implemented to minimize the risks associated with continued resource consumption (DoS attacks), e.g., imposing a limit on the number of P-MAARs associated with a given MN.

The CMD and the participating MAARs MUST be trusted parties authorized perform all operations relevant to their role.

There are some privacy considerations to consider. While the involved parties trust each other, the signaling involves disclosing information about the previous locations visited by each MN, as well as the active prefixes they are using at a given point of time. Therefore, mechanisms MUST be in place to ensure that MAARs and CMDs do not disclose this information to other parties or use it for other ends than providing the distributed mobility support specified in this document.

7. References

7.1. Normative References

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

[RFC4191] Draves, R. and D. Thaler, "Default Router Preferences and More-Specific Routes", RFC 4191, DOI 10.17487/RFC4191, November 2005, <https://www.rfc-editor.org/info/rfc4191>.

[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>.

[RFC5213] Gundavelli, S., Ed., Leung, K., Devarapalli, V., Chowdhury, K., and B. Patil, "Proxy Mobile IPv6", RFC 5213, DOI 10.17487/RFC5213, August 2008, <https://www.rfc-editor.org/info/rfc5213>.

[RFC6275] Perkins, C., Ed., Johnson, D., and J. Arkko, "Mobility Support in IPv6", RFC 6275, DOI 10.17487/RFC6275, July 2011, <https://www.rfc-editor.org/info/rfc6275>.

[RFC7333] Chan, H., Ed., Liu, D., Seite, P., Yokota, H., and J. Korhonen, "Requirements for Distributed Mobility Management", RFC 7333, DOI 10.17487/RFC7333, August 2014, <https://www.rfc-editor.org/info/rfc7333>.

[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>.

7.2. Informative References

[DISTRIBUTED-ANCHORING] Bernardos, C. and J. Zuniga, "PMIPv6-based distributed anchoring", Work in Progress, Internet-Draft, draft-bernardos-dmm-distributed-anchoring-09, 29 May 2017, <https://tools.ietf.org/html/draft-bernardos-dmm-distributed-anchoring-09>.

[DMM-PMIP] Bernardos, C., Oliva, A., and F. Giust, "A PMIPv6-based solution for Distributed Mobility Management", Work in Progress, Internet-Draft, draft-bernardos-dmm-pmip-09, 8 September 2017, <https://tools.ietf.org/html/draft-bernardos-dmm-pmip-09>.

[RFC6724] Thaler, D., Ed., Draves, R., Matsumoto, A., and T. Chown, "Default Address Selection for Internet Protocol Version 6 (IPv6)", RFC 6724, DOI 10.17487/RFC6724, September 2012, <https://www.rfc-editor.org/info/rfc6724>.

[RFC7429] Liu, D., Ed., Zuniga, JC., Ed., Seite, P., Chan, H., and CJ. Bernardos, "Distributed Mobility Management: Current Practices and Gap Analysis", RFC 7429, DOI 10.17487/RFC7429, January 2015, <https://www.rfc-editor.org/info/rfc7429>.

[RFC8563] Katz, D., Ward, D., Pallagatti, S., Ed., and G. Mirsky, Ed., "Bidirectional Forwarding Detection (BFD) Multipoint Active Tails", RFC 8563, DOI 10.17487/RFC8563, April 2019, <https://www.rfc-editor.org/info/rfc8563>.

Acknowledgements

The authors would like to thank Dirk von Hugo, John Kaippallimalil, Ines Robles, Joerg Ott, Carlos Pignataro, Vincent Roca, Mirja Khlewind, ric Vyncke, Adam Roach, Benjamin Kaduk, and Roman Danyliw for the comments on this document. The authors would also like to thank Marco Liebsch, Dirk von Hugo, Alex Petrescu, Daniel Corujo, Akbar Rahman, Danny Moses, Xinpeng Wei, and Satoru Matsushima for their comments and discussion on the documents [DISTRIBUTED-ANCHORING] and [DMM-PMIP], on which the present document is based.

The authors would also like to thank Lyle Bertz and Danny Moses for their in-depth review of this document and their very valuable comments and suggestions.

Authors' Addresses

Carlos J. Bernardos
Universidad Carlos III de Madrid
Av. Universidad, 30
28911 Leganes Madrid
Spain

Phone: +34 91624 6236
Email: cjbc@it.uc3m.es
URI: http://www.it.uc3m.es/cjbc/

Antonio de la Oliva
Universidad Carlos III de Madrid
Av. Universidad, 30
28911 Leganes Madrid
Spain

Phone: +34 91624 8803
Email: aoliva@it.uc3m.es
URI: http://www.it.uc3m.es/aoliva/

Fabio Giust
Athonet S.r.l.
via Ca' del Luogo 6/8
36050 Bolzano Vicentino (VI)
Italy

Email: fabio.giust.research@gmail.com

Juan Carlos Ziga
SIGFOX
425 rue Jean Rostand
31670 Labege
France

Email: j.c.zuniga@ieee.org
URI: http://www.sigfox.com/

Alain Mourad
InterDigital Europe

Email: Alain.Mourad@InterDigital.com
URI: http://www.InterDigital.com/