DMM Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 12, 2014

CJ. Bernardos
A. de la Oliva
UC3M
F. Giust
IMDEA Networks and UC3M

July 11, 2013

# A PMIPv6-based solution for Distributed Mobility Management
draft-bernardos-dmm-pmip-03

## Abstract

The number of mobile users and their traffic demand is expected to be ever-increasing in future years, and this growth can represent a limitation for deploying current mobility management schemes that are intrinsically centralized, e.g., Mobile IPv6 and Proxy Mobile IPv6. For this reason it has been waved a need for distributed and dynamic mobility management approaches, with the objective of reducing operators' burdens, evolving to a cheaper and more efficient architecture.

This draft describes multiple solutions for network-based distributed mobility management inspired by the well known Proxy Mobile IPv6.

## Requirements Language

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].

## Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 12, 2014.

Table of Contents



## 1. Introduction

Current IP mobility solutions, standardized with the names of Mobile IPv6 [RFC6275], or Proxy Mobile IPv6 [RFC5213], just to cite the two most relevant examples, offer mobility support at the cost of handling operations at a cardinal point, the mobility anchor, and burdening it with data forwarding and control mechanisms for a great amount of users. As stated in [I-D.ietf-dmm-requirements], 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 on the mobility management service (i.e., mobility is offered on a per-node basis, not being possible to define finer granularity policies, as for example per-application).

The purpose of Distributed Mobility Management is to overcome the limitations of the traditional centralized mobility management [I-D.ietf-dmm-requirements] [I-D.ietf-dmm-best-practices-gap-analysis]; the main concept behind DMM solutions is indeed bringing the mobility anchor closer to the MN. Following this idea, in our proposal, the central anchor is moved to the edge of the network, being deployed in the default gateway of the mobile node. That is, the first elements that provide IP connectivity to a set of MNs are also the mobility managers for those MNs. In the following, we will call these entities Mobility Anchor and Access Routers (MAARs).

This document focuses on network-based DMM, hence the starting point is making PMIPv6 working in a distributed manner [I-D.ietf-dmm-best-practices-gap-analysis]. In our proposal, as in PMIPv6, mobility is handled by the network without the MNs involvement, but, differently from PMIP, when the MN moves from one access network to another, it also changes anchor router, 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 they are routable at that MAAR.

In the following, we consider two main approaches to design our DMM solutions:

- Partially distributed schemes, where the data plane only is distributed among access routers similar to MAGs, whereas the control plane is kept centralized towards a cardinal node used as information store, but relieved from any route management and MN’s data forwarding task.

- o Fully distributed schemes, where both data and control planes are distributed among the access routers.

## 2. Terminology

The following terms used in this document are defined in the Proxy Mobile IPv6 specification [RFC5213]:

Local Mobility Anchor (LMA)

Mobile Access Gateway (MAG)

Mobile Node (MN)

Binding Cache Entry (BCE)

Proxy Care-of Address (P-CoA)

Proxy Binding Update (PBU)

Proxy Binding Acknowledgement (PBA)

The following terms are defined and used in this document:

MAAR (Mobility Anchor and Access Router). First hop router where the mobile nodes attach to. It also plays the role of mobility manager for the IPv6 prefixes it anchors, running the functionalities of PMIP's MAG and LMA.

CMD (Central Mobility Database). Node that stores the BCEs allocated for the MNs in the mobility domain.

P-MAAR (Previous MAAR). MAAR which was previously visited by the MN and is still involved in an active flow using an IPv6 prefix it has advertised to the MN (i.e., MAAR where that IPv6 prefix is anchored). There might be multiple P-MAARs for an MN's mobility session.

S-MAAR (Serving MAAR). MAAR which the MN is currently attached to.

## 3. Partially distributed solution

The following solution consists in de-coupling the entities that participates 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 on the MAARs, relies on a central entity called Central Mobility Database (CMD). In the proposed architecture, the hierarchy present in PMIP between LMA and MAG is preserved, but with the following substantial variations:

- The LMA is relieved from the data forwarding role, only the Binding Cache and its management operations are maintained. Hence the LMA is renamed into Central Mobility Database (CMD). 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 have to be extended to include information regarding previous MAARs where the mobile node was anchored and still retains active data sessions, see Appendix B for further details.

- 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 on the Central Mobility Database (CMD) to access and update information related to the MNs, stored as mobility sessions; hence, a centralized node maintains a global view on the status of the network. The CMD is queried whenever a MN is detected to join/leave 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 messages exchange sequence might vary to optimize the update latency and/or the signaling overhead. Here is presented one method for the initial registration, and three different approaches to update the mobility sessions using PBUs and PBAs. 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.

### 3.1. Initial registration

Upon the MN's attachment to a MAAR, say MAAR1, if the MN is authorized for the service, an IPv6 global prefix belonging to the MAAR's prefix pool is reserved for it (Pref1) into a temporal Binding Cache Entry (BCE) allocated locally. The prefix is sent in a [RFC5213] PBU with the MN's Identifier (MN-ID) to the CMD, which, since the session is new, stores a permanent BCE containing as main fields the MN-ID, the MN's prefix and MAAR1's address as Proxy-CoA. The CMD replies to MAAR1 with a PBA including the usual options defined in PMIP/RFC5213, meaning that the MN's registration is fresh and no past status is available. MAAR1 definitely stores the temporal BCE previously allocated and unicasts a Router Advertisement (RA) to the MN including the prefix reserved before, that can be used by the MN to configure an IPv6 address (e.g., with stateless auto-configuration). The address is routable at the MAAR, in the sense that it is on the path of packets addressed to the MN. Moreover, the MAAR acts as plain router for those packets, as no encapsulation nor special handling takes place. Figure 1 illustrates this scenario.

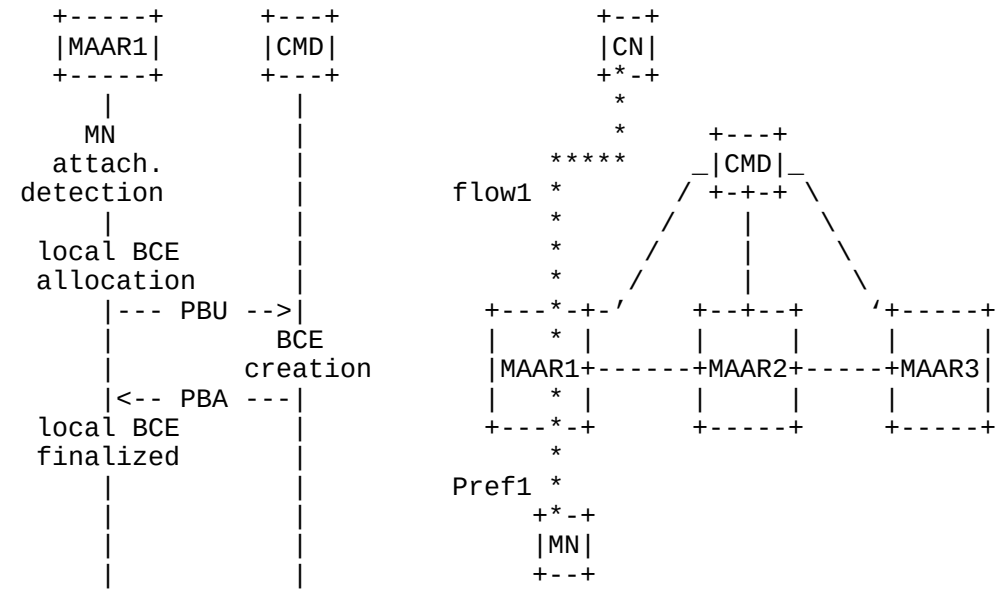


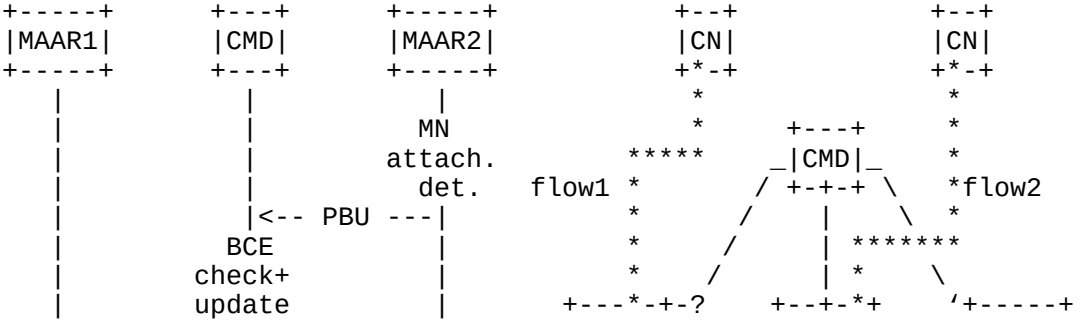

Figure 1: First attachment to the network

### 3.2. The CMD as PBU/PBA relay

When the MN moves from its current access and associates to MAAR2 (now the S-MAAR), MAAR2 reserves another IPv6 prefix (Pref2), it stores a temporal BCE, and it sends a plain PBU to the CMD for registration. Upon PBU reception and BC lookup, the CMD retrieves an already existing entry for the MN, binding the MN-ID to its former location; thus, the CMD forwards the PBU to the MAAR indicated as Proxy CoA (MAAR1), including a new mobility option to communicate the S-MAAR's global address to MAAR1, defined as Serving MAAR Option in Section 3.6.2. The CMD updates the P-CoA field in the BCE related to the MN with the S-MAAR's address.

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, containing the prefix anchored at MAAR1 in the Home Network Prefix option. The CMD, after receiving the PBA, updates the BCE populating 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 (see Appendix B for further details). Also, the CMD send a PBA to the new S-MAAR, containing the previous Proxy-CoA and the prefix anchored to it embedded into a new mobility option called Previous MAAR Option (defined in Section 3.6.1), so that, upon PBA arrival, a bi-directional tunnel can be established between the two MAARs and new routes are set appropriately to recover the IP flow(s) carrying Pref1.

Now packets destined to Pref1 are first received by MAAR1, encapsulated into the tunnel and forwarded to MAAR2, which finally delivers them to their destination. In uplink, when the MN transmits packets using Pref1 as source address, they are sent to MAAR2, as it is MN's new default gateway, then tunneled to MAAR1 which routes them towards the next hop to destination. Conversely, packets carrying Pref2 are routed by MAAR2 without any special packet handling both for uplink and downlink. The procedure is depicted in Figure 2.



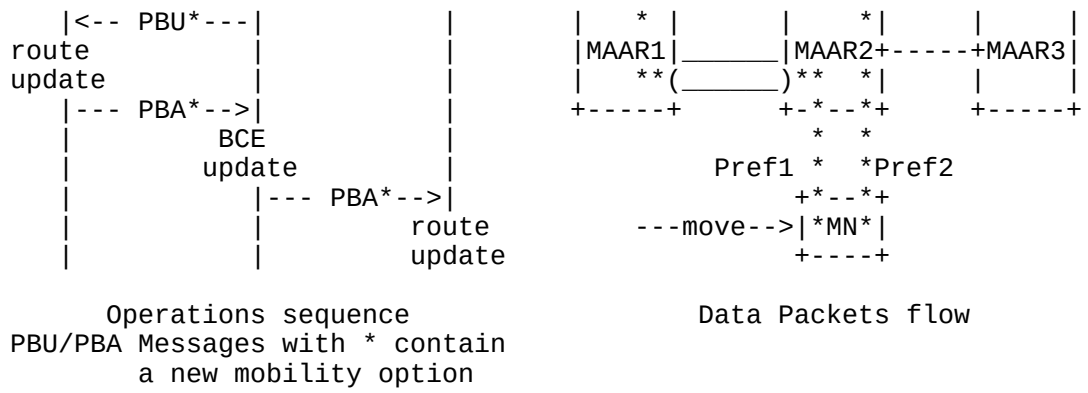



Figure 2: Scenario after a handover, CMD as relay

For next MN's movements the process is repeated except for the number of P-MAARs involved, that rises accordingly 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 as current P-CoA (i.e., the MAAR prior to handover) and in the P-MAARs list. They reply with a PBA to the CMD, which aggregates them into a single one to notify the S-MAAR, that 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 to 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 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.

### 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 signal directly their information to the new S-MAAR. This procedure reflect what was described in Section 3.2 up to the moment the P-MAAR receives the PBU with the P-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 S-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 HNP option and the P-MAAR's address OS taken from the message's source address. The CMD is relieved 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. In Figure 3 is illustrated the new messages sequence, while the data forwarding is unaltered.

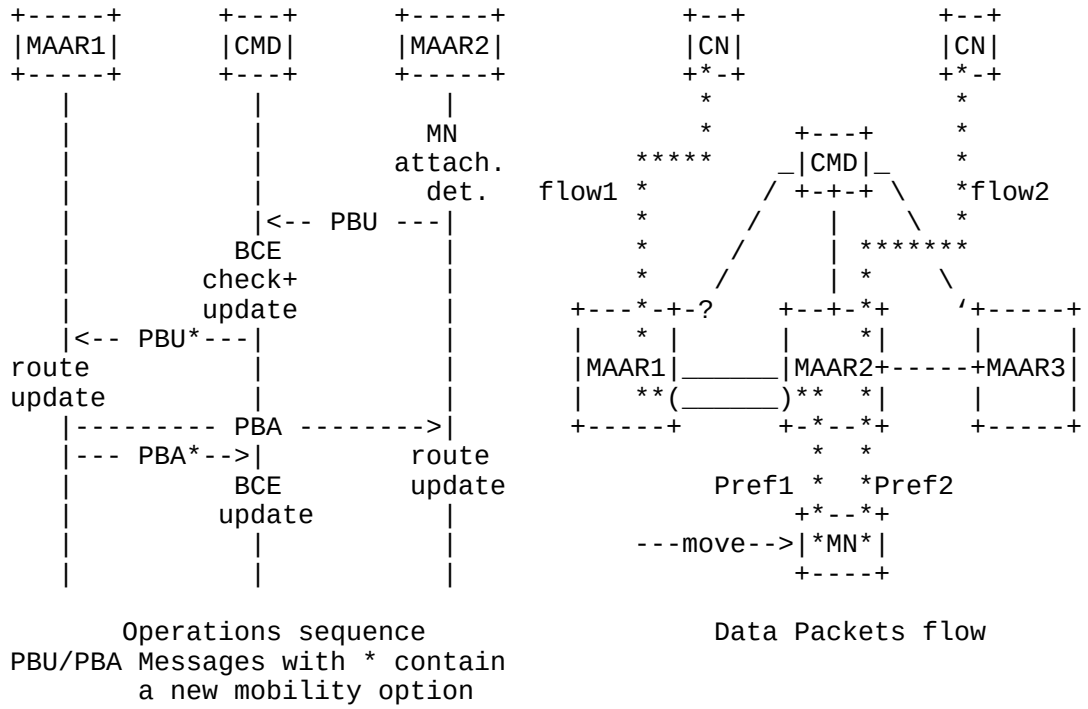



Figure 3: Scenario after a handover, CMD as locator

## 3.4. The CMD as MAAR 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 possess 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 also in this case the P-MAAR option. In parallel, a PBU is sent by the CMD to the P-MAARs containing the S-MAAR option, to notify them about the new MN's location, so 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 is concluded and the information are updated in all network nodes. This procedure is obtained from the first one re-arranging 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.

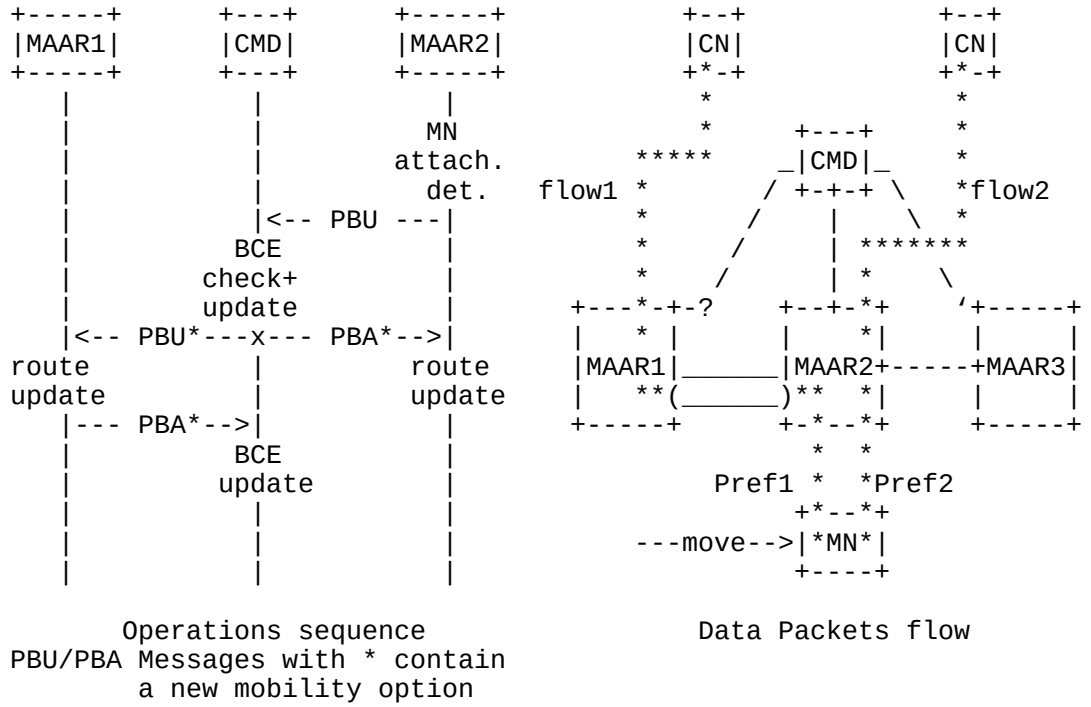



Figure 4: Scenario after a handover, CMD as proxy

## 3.5. De-registration

The de-registration mechanism devised for PMIPv6 is no longer valid in the Partial DMM architecture. This is motivated by the fact that each MAAR handles an independent mobility session (i.e., a single or a set of prefixes) for a given MN, whereas the aggregated session is stored at the CMD. Indeed, when a previous 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 that (those) 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 an old MAAR to de-register a prefix, that is, allowing only a serving MAAR to de-register the whole MN session. This can be achieved by first removing any layer-2 detachment event, so that de-registration is triggered only when the session 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:

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

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

The evaluation of these methods is left for future work.

### 3.6. Message Format

This section defines two Mobility Options to be used in the PBU and PBA messages:

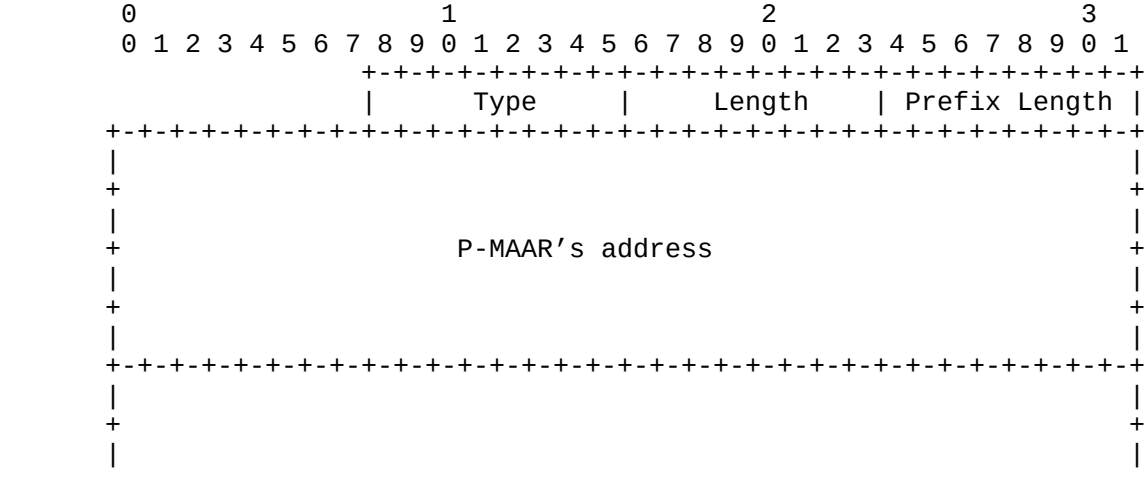
Previous MAAR Option;

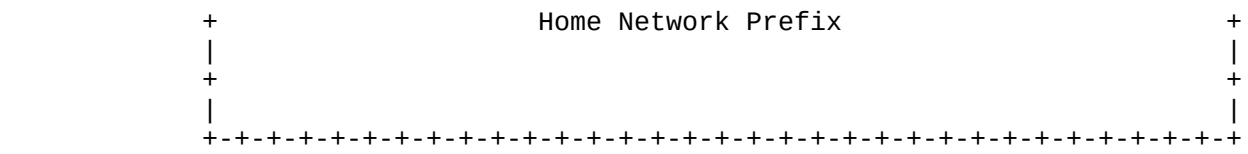
Serving MAAR Option.

In the current draft the messages reflect IPv6 format only. IPv4 compatibility will be added in next release.

#### 3.6.1. Previous MAAR Option

This new option is defined for use with the Proxy Binding Acknowledgement messages exchanged by the CMD to a MAAR. This option is used to notify the S-MAAR about the previous MAAR's global address and the prefix anchored to it. There can be multiple Previous MAAR options present in the message. Its format is as follows:






Type

To be assigned by IANA.

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.

Prefix Length

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

Previous MAAR's address

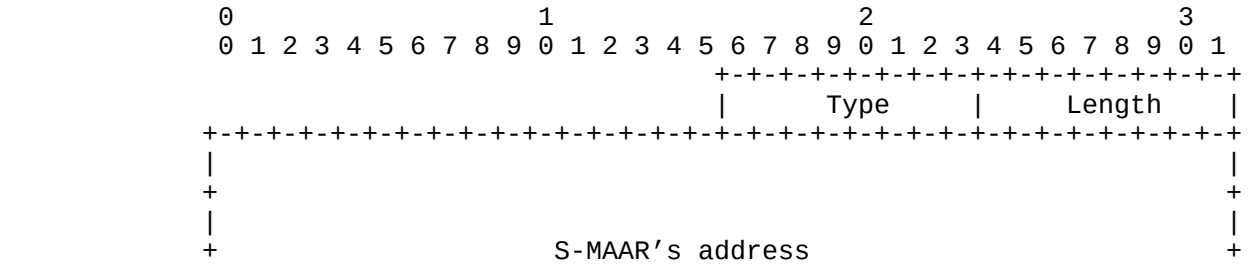
A sixteen-byte field containing the P-MAAR's IPv6 global address.

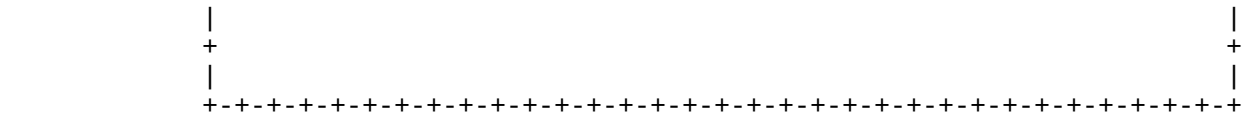
Home Network Prefix

A sixteen-byte field containing the mobile node's IPv6 Home Network Prefix.

### 3.6.2. Serving MAAR Option

This new option is defined for use with the Proxy Binding Update and Proxy Binding Acknowledgement messages exchanged between the CMD and a Previous MAAR. This option is used to notify the P-MAAR about the current Serving MAAR's global address. Its format is as follows:





Type

To be assigned by IANA.

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's address

A sixteen-byte field containing the S-MAAR's IPv6 global address.

## 4. Fully distributed solution

In this section we introduce the guidelines to evolve our partially DMM solution into a fully distributed one. We list the key concepts in the following (some of the points are already enforced in previous sections of this document):

- All MAARs have a pool of global routable IPv6 prefixes to be assigned to MNs on the access link.
- Any central control entity is removed from the architecture and each MAAR will retain it's own cache for the mobile nodes directly anchored to it.
- Both control and data planes are now entirely handled by the MAARs.

Because we aim for a fully distributed approach, the lack of knowledge of other MAARs and their advertised prefixes becomes a serious obstacle. In this particular case, when a MN attaches to a MAAR, there are two main pieces ofi nformation that this MAAR requires to know, to properly assure a mobile node's mobility and continuity of its data flows: i) if the node has any P-MAARs and their addresses; ii) if it has P-MAARs, which prefixes were advertised by which MAAR.

There are several methods to achieve this:

- Make before approaches, employing Layer 2 or Layer 3 mechanisms. The target MAAR is known in advance by the current MAAR before handover, hence the mobility context can be transferred.

- Distributed schemes for MAAR discovery: it can based on a peer-to-peer approach; or it can employ a unicast, multicast or broadcast query system.

- Explicit notification by the MN. For example, extending the layer three IP address configuration mechanisms (e.g., ND).

- Other MN to MAAR communication protocol (e.g., IEEE 802.21).

## 5. IANA Considerations

TBD.

## 6. Security Considerations

The solution assumes that the nodes are trusted and secure MAAR-to-MAAR communications are in place, for instance re-using the security mechanisms defined for PMIPv6. Thus, the solution does not introduce any new security vulnerability.

## 7. Acknowledgments

The authors would like to thank Marco Liebsch for his comments and discussion on this document.

## 8. References

### 8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC5213] Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., and B. Patil, "Proxy Mobile IPv6", RFC 5213, August 2008.

[RFC6275] Perkins, C., Johnson, D., and J. Arkko, "Mobility Support in IPv6", RFC 6275, July 2011.

### 8.2. Informative References

[I-D.bernardos-dmm-distributed-anchoring]
Bernardos, C. and J. Zuniga, "PMIPv6-based distributed anchoring", draft-bernardos-dmm-distributed-anchoring-02 (work in progress), April 2013.

[I-D.ietf-dmm-best-practices-gap-analysis]
Liu, D., Zuniga, J., Seite, P., Chan, A., and C. Bernardos, "Distributed Mobility Management: Current practices and gap analysis", draft-ietf-dmm-best-practices-gap-analysis-01 (work in progress), June 2013.

[I-D.ietf-dmm-requirements]
Chan, A., Liu, D., Seite, P., Yokota, H., and J. Korhonen, "Requirements for Distributed Mobility Management", draft-ietf-dmm-requirements-05 (work in progress), June 2013.

[RFC4620] Crawford, M. and B. Haberman, "IPv6 Node Information Queries", RFC 4620, August 2006.

## Appendix A. Comparison with Requirement document

In this section we descrbe how our solution addresses the DMM requirements listed in [I-D.ietf-dmm-requirements].

### A.1. Distributed Processing

"IP mobility, network access and routing solutions provided by DMM MUST enable distributed processing for mobility management of some flows so that traffic does not need to traverse centrally deployed mobility anchors and thereby avoid non-optimal routes."

In our solution, a MAAR is responsible to handle the mobility for those IP flows started when the MN is attached to it. As long as the MN remains connected to the MAAR's access links, the IP packets of such flows can benefit from the optimal path. When the MN moves to another MAAR, the path becomes non-optimal for ongoing flows, as they are anchored to the previous MAAR, but newly started IP sessions are forwarded by the new MAAR through the optimal path.

### A.2. Transparency to Upper Layers when needed

"DMM solutions MUST provide transparent mobility support above the IP layer when needed. Such transparency is needed, for example, when, upon change of point of attachment to the network, an application flow cannot cope with a change in the IP address. However, it is not always necessary to maintain a stable home IP address or prefix for every application or at all times for a mobile node."

Our DMM solution operates at the IP layer, hence upper layers are totally transparent to the mobility operations. In particular, ongoing IP sessions are not disrupted after a change of access network. The routability of the old address is ensured by the IP tunnel with the old MAAR. New IP sessions are started with the new address. From the application's perspective, those processes which sockets are bound to a unique IP address do not suffer any impact. For the other applications, the sockets bound to the old address are preserved, whereas next sockets use the new address.

### A.3. IPv6 deployment

"DMM solutions SHOULD target IPv6 as the primary deployment environment and SHOULD NOT be tailored specifically to support IPv4, in particular in situations where private IPv4 addresses and/or NATs are used."

The DMM solution we propose targets IPv6 only.

### A.4. Existing mobility protocols

"A DMM solution SHOULD first consider reusing and extending IETF-standardized protocols before specifying new protocols."

This DMM solution is derived from the operations and messages specified in [RFC5213].

### A.5. Co-existence with deployed networks and hosts

"The DMM solution MUST be able to co-exist with existing network deployments and end hosts. For example, depending on the environment in which DMM is deployed, DMM solutions may need to be compatible with other deployed mobility protocols or may need to co-exist with a network or mobile hosts/routers that do not support DMM protocols. The mobile node may also move between different access networks, where some of them may support neither DMM nor another mobility protocol. Furthermore, a DMM solution SHOULD work across different networks, possibly operated as separate administrative domains, when allowed by the trust relationship between them."

The partially DMM solution can be extended to provide a fallback mechanism to operate as legacy Proxy Mobile IPv6. It is necessary to instruct MAARs to always establish a tunnel with the same MAAR, working as LMA. The fully DMM solution can be extended as well, but it requires more intervention. The partially DMM solution can be deployed across different domains with trust agreements if the CMDs ot the operators are enabled to transfer context from one node to another. The fully DMM solution works across multiple domains if both solution apply the same signalling scheme.

### A.6. Security considerations

"DMM protocol solutions MUST consider security risks introduced by DMM into the network. Such considerations may include authentication and authorization mechanisms that allow a mobile host/router to use the mobility support provided by the DMM solution; measures against redirecting traffic to the wrong host when providing DMM support; signaling message protection for authentication, integrity and confidentiality."

The proposed solution does not specify a security mechanism, given that the same mechanism for PMIPv6 can be used.

### A.7. Multicast

"DMM SHOULD enable multicast solutions in flexible distribution scenario. This flexibility pertains to the preservation of IP multicast nature from the perspective of a mobility entity and transmission of multicast packets to/from various multicast- enabled entities. Therefore, this flexibility enables different IP multicast flows with respect to a mobile host to be managed (e.g., subscribed, received and/or transmitted) using multiple multicast-enabled endpoints."

This solution in its current version does not specify any support for multicast traffic, which is left for study in future versions.

## Appendix B. Implementation experience

The solution described in section Section 3.4 has been implemented in a real test-bed comprising 3 MAARs, one CMD, one MN and a CN. The CN is connected to the DMM domain through a router, which simulates the gateway to the internet cloud. All the machines used are Linux UBUNTU 10.04 systems with kernel 2.26.32, but the system has been tested also under newer platforms.

The code is developed in ANSI C from the existing UMIP implementation for PMIP within the framework of the MEDIEVAL EU project. The most

relevant changes with respect to the original version are related to how to create the CMD and MAAR's state machines from those of an LMA and a MAG; for this purpose, part of the LMA code was copied to the MAG, in order to send PBA messages and parse PBU. Also, the LMA routing functions were removed completely, and moved to the MAG, because MAARs need to route through the tunnels in downlink (as an LMA) and in uplink (as a MAG).

Tunnel management is hence a relevant technical aspect, as multiple tunnels are established by a single MAAR, which keeps their status directly into the MN's BCE. Indeed, from the implementation experience it was chosen to create an ancillary data structure as field within a BCE: the data structure is called "MAAR list" and stores the previous MAARs' address and the corresponding prefix(es) assigned for the MN. Only the CMD and the serving MAAR store this data structure, because the CMD maintains the global MN's mobility session formed during the MN's roaming within the domain, and the serving MAAR needs to know which previous MAARs were visited, the prefix(es) they assigned and the tunnels established with them. Conversely, a previous MAAR only needs to know which is the current Serving MAAR and establish a single tunnel with it. For this reason, a MAAR that receives a PBU from the CMD (meaning that the MN attached to another MAAR), first sets up the routing state for the MN's prefix(es) it is anchoring, then stop the BCE expiry timer and deletes the MAAR list (if present) since it is no longer useful.

In order to have the MN totally unaware of the changes in the access link, all MAARs exhibit the same L2 and L3 identifiers in the access interface (as the PMIPv6 Fixed MAG Link Local Address feature). A solution is under study to avoid this configuration and influence the MN on the source address choice. Moreover, it should be noted that the protocols designed in the document work only at the network layer to handle the MNs joining or leaving the domain. This should guarantee a certain independency to a particular access technology. The implementation reflects this reasoning, but we argue that an interaction with lower layers produces a more effective attachment and detachment detection, therefore improving the performance, also regarding de-registration mechanisms.

It was chosen to implement the "proxy" solution because it produces the shortest handover latency, but a slight modification on the CMD state machine can produce the first scenario described ("relay") which guarantees a more consistent request/ack scheme between the MAARS. By modifying also the MAAR's state machine it can be implemented the second solution ("locator").

A prototype implementation of the solution described in this document, together with the distributed logical interface concept

specified in [I-D.bernardos-dmm-distributed-anchoring] was shown during IETF 83rd, in Paris in March 2012. An enhancement version of this demo is going to be presented at the 87th IETF meeting in Berlin, July 2013. The new demo includes a use case scenario employing a CDN system for video delivery.

## Authors' Addresses

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

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

Fabio Giust
Institute IMDEA Networks and Universidad Carlos III de Madrid
Av. del Mar Mediterraneo, 22
Leganes, Madrid 28918
Spain

Phone: +34 91481 6979
Email: fabio.giust@imdea.org