Distributed Mobility Management [dmm]
Internet-Draft
Expires: December 11, 2016

C. Perkins
Futurewei
V. Devarapalli
Vasona Networks

June 9, 2016

MN Identifier Types for RFC 4283 Mobile Node Identifier Option
draft-ietf-dmm-4283mnids-02.txt

Abstract

Additional Identifier Types are proposed for use with the Mobile Node Identifier Option for MIPv6 (RFC 4283).

Status of This Memo

This Internet-Draft will expire on December 11, 2016.

Copyright Notice

Table of Contents

1. Introduction

The Mobile Node Identifier Option for MIPv6 [RFC4283] has proved to be a popular design tool for providing identifiers for mobile nodes during authentication procedures with AAA protocols such as Diameter [RFC3588]. To date, only a single type of identifier has been specified, namely the MN NAI. Other types of identifiers are in common use, and even referenced in RFC 4283. In this document, we propose adding some basic types that are defined in various telecommunications standards, including types for IMSI [ThreeGPP-IDS], P-TMSI [ThreeGPP-IDS], IMEI [ThreeGPP-IDS], and GUTI [ThreeGPP-IDS]. In addition, we specify the IPv6 address itself and IEEE MAC-layer addresses as mobile node identifiers. Defining identifiers that are tied to the physical elements of the device (RFID, MAC address etc.) help in deployment of Mobile IP because in many cases such identifiers are the most natural means for uniquely identifying the device, and will avoid additional look-up steps that might be needed if other identifiers were used.

2. New Mobile Node Identifier Types

The following types of identifiers are commonly used to identify mobile nodes. For each type, references are provided with full details on the format of the type of identifer.

The Tag Data standard promoted by Electronic Product Code(TM) (abbreviated EPC) supports several encoding systems or schemes including

- RFID-GID (Global Identifier),
- RFID-SGTIN (Serialized Global Trade Item Number),
- RFID-SSCC (Serial Shipping Container),
- RFID-SGLN (Global Location Number),
- RFID-GRAI (Global Returnable Asset Identifier),
- RFID-DOD (Department of Defense ID), and
- RFID-GIAI (Global Individual Asset Identifier).

For each RFID scheme except GID, there are two variations: a 64-bit scheme (for example, SGLN-64) and a 96-bit scheme (SGLN-96). GID has only a 96-bit scheme. Within each scheme, an EPC identifier can be represented in a binary form or other forms such as URI.

The following list includes the above RFID types as well as various other common identifiers and several different types of DUIDs.

Mobile Node Identifier Description

Identifier Type	Description	Reference
IPv6 Address		[RFC4291]
IMSI	International Mobile Subscriber Identity	[ThreeGPP-IDS]
P-TMSI	Packet-Temporary Mobile Subscriber Identity	[ThreeGPP-IDS]
GUTI	Globally Unique Temporary ID	[ThreeGPP-IDS]
EUI-48 address	48-bit Extended Unique Identifier	[IEEE802]
EUI-64 address	64-bit Extended Unique Identifier-64 bit	[IEEE802]

DUID-LLT	DHCPv6 Unique Identifier: Link-Layer address plus timestamp	[RFC3315]
DUID-EN	DHCPv6 Unique Identifier: Enterprise Number plus add’l data	[RFC3315]
DUID-LL	DHCPv6 Unique Identifier: Link-Layer address	[RFC3315]
DUID-UUID	DHCPv6 Unique Identifier: other conformant format	[RFC6355]
RFID-SGTIN-64	64-bit Serialized Global Trade Item Number	[EPC-Tag-Data]
RFID-SSCC-64	64-bit Serial Shipping Container	[EPC-Tag-Data]
RFID-SGLN-64	64-bit Serialized Global Location Number	[EPC-Tag-Data]
RFID-GRAI-64	64-bit Global Returnable Asset Identifier	[EPC-Tag-Data]
RFID-DOD-64	64-bit Department of Defense ID	[RFID-DoD-spec]
RFID-GIAI-64	64-bit Global Individual Asset Identifier	[EPC-Tag-Data]
RFID-GID-96	96-bit Global Identifier	[EPC-Tag-Data]
RFID-SGTIN-96	96-bit Serialized Global Trade Item Number	[EPC-Tag-Data]
RFID-SSCC-96	96-bit Serial Shipping Container	[EPC-Tag-Data]
RFID-SGLN-96	96-bit Serialized Global Location Number	[EPC-Tag-Data]
RFID-GRAI-96	96-bit Global Returnable Asset Identifier	[EPC-Tag-Data]
RFID-DOD-96	96-bit Department of Defense ID	[RFID-DoD-spec]
RFID-GIAI-96	96-bit Global Individual Asset Identifier	[EPC-Tag-Data]
RFID-GID-URI	Global Identifier represented as URI	[EPC-Tag-Data]
RFID-SGTIN-URI	Serialized Global Trade Item Number represented as URI	[EPC-Tag-Data]
RFID-SSCC-URI	Serial Shipping Container represented as URI	[EPC-Tag-Data]
RFID-SGLN-URI	Global Location Number represented as URI	[EPC-Tag-Data]
RFID-GRAI-URI	Global Returnable Asset Identifier represented as URI	[EPC-Tag-Data]
RFID-DOD-URI	Department of Defense ID represented as URI	[RFID-DoD-spec]
RFID-GIAI-URI	Global Individual Asset	[EPC-Tag-Data]

	Identifier represented as URI	

Table 1

3. Descriptions of MNID types

In this section descriptions for the various MNID types are provided.

3.1. Description of the IPv6 address type

The IPv6 address [RFC4291] is encoded as a 16 octet string containing the full IPv6 address.

3.2. Description of the IMSI MNID type

The International Mobile Subscriber Identity (IMSI) [ThreeGPP-IDS] is at most 15 decimal digits (i.e., digits from 0 through 9). The IMSI MUST be encoded as a string of octets in network order, where each digit occupies 4 bits. The last digit MUST be zero padded, if needed, for full octet size. For example an example IMSI 123456123456789 would be encoded as follows:

```
   0x12, 0x34, 0x56, 0x12, 0x34, 0x56, 0x78, 0x90
```

3.3. Description of the EUI-48 address type

The IEEE EUI-48 address [IEEE802-eui48] is encoded as a 6 octet string containing the IEEE EUI-48 address.

3.4. Description of the EUI-64 address type

The IEEE EUI-64 address [IEEE802-eui64] is encoded as a 8 octet string containing the full IEEE EUI-64 address.

3.5. Description of the DUID-LLT type

The DUID-LLT is the DHCPv6 Unique Identifier (DUID) formulated by concatenating the link-layer address plus a timestamp [RFC3315]. This type of DUID consists of a two octet type field containing the value 1, a two octet hardware type code, four octets containing a time value, followed by link-layer address of any one network interface that is connected to the DHCP device at the time that the DUID is generated. The time value is the time that the DUID is generated represented in seconds since midnight (UTC), January 1, 2000, modulo 2^32. Since the link-layer address can be of variable length [RFC2464], the DUID-LLT is of variable length.

3.6. Description of the DUID-EN type

The DUID-EN is the DHCPv6 Unique Identifier (DUID) formulated by concatenating the Enterprise Number plus some additional data [RFC3315]. This form of DUID is assigned by the vendor to the device. It consists of a two octet type field containing the value 2, the vendor’s registered Private Enterprise Number as maintained by IANA, followed by a unique identifier assigned by the vendor. Since the vendor’s unique identifier can be of variable length, the DUID-EN is of variable length.

3.7. Description of the DUID-LL type

The DUID-LL is the DHCPv6 Unique Identifier (DUID) formulated by concatenating the network hardware type code and the link-layer address [RFC3315]. This type of DUID consists of two octets containing the DUID type 3, a two octet network hardware type code, followed by the link-layer address of any one network interface that is permanently connected to the client or server device. For example, a host that has a network interface implemented in a chip that is unlikely to be removed and used elsewhere could use a DUID-LL. Since the link-layer address can be of variable length, the DUID-LL is of variable length.

3.8. Description of the DUID-UUID type

The DUID-UUID [RFC6355] is the DHCPv6 Unique Identifier based on the Universally Unique IDentifier (UUID) [RFC4122]. This type of DUID consists of two octets containing the DUID type 4, followed by 128-bit UUID.

3.9. Description of the RFID types

The General Identifier (GID) that is used with RFID is composed of three fields - the General Manager Number, Object Class and Serial Number. The General Manager Number identifies an organizational entity that is responsible for maintaining the numbers in subsequent fields. GID encodings include a fourth field, the header, to guarantee uniqueness in the namespace defined by EPC.

Some of the RFID types depend on the Global Trade Item Number (GTIN) code defined in the General EAN.UCC Specifications [EANUCCGS]. A GTIN identifies a particular class of object, such as a particular kind of product or SKU.

The EPC encoding scheme for SGTIN permits the direct embedding of EAN.UCC System standard GTIN and Serial Number codes on EPC tags. In all cases, the check digit is not encoded. Two encoding schemes are specified, SGTIN-64 (64 bits) and SGTIN-96 (96 bits).

The Serial Shipping Container Code (SSCC) is defined by the EAN.UCC Specifications. Unlike the GTIN, the SSCC is already intended for assignment to individual objects and therefore does not require additional fields to serve as an EPC pure identity. Two encoding schemes are specified, SSCC-64 (64 bits) and SSCC-96 (96 bits).

The Global Location Number (GLN) is defined by the EAN.UCC Specifications. A GLN can represent either a discrete, unique physical location such as a warehouse slot, or an aggregate physical location such as an entire warehouse. In addition, a GLN can represent a logical entity that performs a business function such as placing an order. The Serialized Global Location Number (SGLN) includes the Company Prefix, Location Reference, and Serial Number.

The Global Returnable Asset Identifier is (GRAI) is defined by the General EAN.UCC Specifications. Unlike the GTIN, the GRAI is already intended for assignment to individual objects and therefore does not require any additional fields to serve as an EPC pure identity. The GRAI includes the Company Prefix, Asset Type, and Serial Number.

The Global Individual Asset Identifier (GIAI) is defined by the General EAN.UCC Specifications. Unlike the GTIN, the GIAI is already intended for assignment to individual objects and therefore does not require any additional fields to serve as an EPC pure identity. The GRAI includes the Company Prefix, and Individual Asset Reference.

The DoD Construct identifier is defined by the United States Department of Defense (DoD). This tag data construct may be used to encode tags for shipping goods to the DoD by a supplier who has already been assigned a CAGE (Commercial and Government Entity) code.

3.9.1. Description of the RFID-SGTIN-64 type

The RFID-SGTIN-64 is encoded as specified in [EPC-Tag-Data]. The SGTIN-64 includes five fields: Header, Filter Value (additional data that is used for fast filtering and pre-selection), Company Prefix Index, Item Reference, and Serial Number. Only a limited number of Company Prefixes can be represented in the 64-bit tag.

3.9.2. Description of the RFID-SGTIN-96 type

The RFID-SGTIN-96 is encoded as specified in [EPC-Tag-Data]. The SGTIN-96 includes six fields: Header, Filter Value, Partition (an indication of where the subsequent Company Prefix and Item Reference numbers are divided), Company Prefix Index, Item Reference, and Serial Number.

3.9.3. Description of the RFID-SSCC-64 type

The RFID-SSCC-64 is encoded as specified in [EPC-Tag-Data]. The SSCC-64 includes four fields: Header, Filter Value, Company Prefix Index, and Serial Reference. Only a limited number of Company Prefixes can be represented in the 64-bit tag.

3.9.4. Description of the RFID-SSCC-96 type

The RFID-SSCC-96 is encoded as specified in [EPC-Tag-Data]. The SSCC-96 includes six fields: Header, Filter Value, Partition, Company Prefix, and Serial Reference, as well as 24 bits that remain Unallocated and must be zero.

3.9.5. Description of the RFID-SGLN-64 type

The RFID-SGLN-64 type is encoded as specified in [EPC-Tag-Data]. The SGLN-64 includes five fields: Header, Filter Value, Company Prefix Index, Location Reference, and Serial Number.

3.9.6. Description of the RFID-SGLN-96 type

The RFID-SGLN-96 type is encoded as specified in [EPC-Tag-Data]. The SGLN-96 includes six fields: Header, Filter Value, Partition, Company Prefix, Location Reference, and Serial Number.

3.9.7. Description of the RFID-GRAI-64 type

The RFID-GRAI-64 type is encoded as specified in [EPC-Tag-Data]. The GRAI-64 includes five fields: Header, Filter Value, Company Prefix Index, Asset Type, and Serial Number.

3.9.8. Description of the RFID-GRAI-96 type

The RFID-GRAI-96 type is encoded as specified in [EPC-Tag-Data]. The GRAI-96 includes six fields: Header, Filter Value, Partition, Company Prefix, Asset Type, and Serial Number.

3.9.9. Description of the RFID-GIAI-64 type

The RFID-GIAI-64 type is encoded as specified in [EPC-Tag-Data]. The GIAI-64 includes four fields: Header, Filter Value, Company Prefix Index, and Individual Asset Reference.

3.9.10. Description of the RFID-GIAI-96 type

The RFID-GIAI-96 type is encoded as specified in [EPC-Tag-Data]. The GIAI-96 includes five fields: Header, Filter Value, Partition, Company Prefix, and Individual Asset Reference.

3.9.11. Description of the RFID-DoD-64 type

The RFID-DoD-64 type is encoded as specified in [RFID-DoD-spec]. The DoD-64 type includes four fields: Header, Filter Value, Government Managed Identifier, and Serial Number.

3.9.12. Description of the RFID-DoD-96 type

The RFID-DoD-96 type is encoded as specified in [RFID-DoD-spec]. The DoD-96 type includes four fields: Header, Filter Value, Government Managed Identifier, and Serial Number.

3.9.13. Description of the RFID URI types

In some cases, it is desirable to encode in URI form a specific encoding of an RFID tag. For example, an application may prefer a URI representation for report preparation. Applications that wish to manipulate any additional data fields on tags may need some representation other than the pure identity forms.

For this purpose, the fields as represented the previous sections are associated with specified fields in the various URI types. For instance, the URI may have fields such as CompanyPrefix, ItemReference, or SerialNumber. For details and encoding specifics, consult [EPC-Tag-Data].

4. Security Considerations

This document does not introduce any security mechanisms, and does not have any impact on existing security mechanisms. Insofar as the selection of a security association may be dependent on the exact form of a mobile node identifier, additional specification may be necessary when the new identifier types are employed with the general AAA mechanisms for mobile node authorizations.

Some identifiers (e.g., IMSI) are considered to be private information. If used in the MNID extension as defined in this document, the packet including the MNID extension should be encrypted so that personal information or trackable identifiers would not be inadvertently disclosed to passive observers. Operators can potentially apply IPsec Encapsulating Security Payload (ESP) with confidentiality and integrity protection for protecting the location information.

Moreover, MNIDs containing sensitive identifiers might only be used for signaling during initial network entry. Subsequent binding update exchanges might then rely on a temporary identifier allocated during the initial network entry, perhaps using mechanisms not standardized within the IETF. Managing the association between long-lived and temporary identifiers is outside the scope of this document.

5. IANA Considerations

The new mobile node identifier types defined in the document should be assigned values from the "Mobile Node Identifier Option Subtypes" registry. The following values should be assigned.

New Mobile Node Identifier Types

Identifier Type	Identifier Type Number
IPv6 Address	2
IMSI	3
P-TMSI	4
EUI-48 address	5
EUI-64 address	6
GUTI	7
DUID-LLT	8
DUID-EN	9
DUID-LL	10
DUID-UUID	11
	12-15 reserved
	16 reserved
RFID-SGTIN-64	17
RFID-SSCC-64	18
RFID-SGLN-64	19
RFID-GRAI-64	20
RFID-DOD-64	21
RFID-GIAI-64	22
	23 reserved
RFID-GID-96	24
RFID-SGTIN-96	25
RFID-SSCC-96	26
RFID-SGLN-96	27
RFID-GRAI-96	28
RFID-DOD-96	29
RFID-GIAI-96	30
	31 reserved
RFID-GID-URI	32
RFID-SGTIN-URI	33
RFID-SSCC-URI	34
RFID-SGLN-URI	35
RFID-GRAI-URI	36
RFID-DOD-URI	37
RFID-GIAI-URI	38
	39-255 reserved

Table 2

See Section 3 for additional information about the identifier types.

6. Acknowledgements

The authors wish to acknowledge Hakima Chaouchi, Jouni Korhonen and Sri Gundavelli for their helpful comments.

7. References

7.1. Normative References

[RFC2464] Crawford, M., "Transmission of IPv6 Packets over Ethernet Networks", RFC 2464, DOI 10.17487/RFC2464, December 1998, <http://www.rfc-editor.org/info/rfc2464>.

[RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July 2003, <http://www.rfc-editor.org/info/rfc3315>.

[RFC4122] Leach, P., Mealling, M., and R. Salz, "A Universally Unique IDentifier (UUID) URN Namespace", RFC 4122, DOI 10.17487/RFC4122, July 2005, <http://www.rfc-editor.org/info/rfc4122>.

[RFC4283] Patel, A., Leung, K., Khalil, M., Akhtar, H., and K. Chowdhury, "Mobile Node Identifier Option for Mobile IPv6 (MIPv6)", RFC 4283, DOI 10.17487/RFC4283, November 2005, <http://www.rfc-editor.org/info/rfc4283>.

[RFC4285] Patel, A., Leung, K., Khalil, M., Akhtar, H., and K. Chowdhury, "Authentication Protocol for Mobile IPv6", RFC 4285, DOI 10.17487/RFC4285, January 2006, <http://www.rfc-editor.org/info/rfc4285>.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, DOI 10.17487/RFC4291, February 2006, <http://www.rfc-editor.org/info/rfc4291>.

[RFC6355] Narten, T. and J. Johnson, "Definition of the UUID-Based DHCPv6 Unique Identifier (DUID-UUID)", RFC 6355, DOI 10.17487/RFC6355, August 2011, <http://www.rfc-editor.org/info/rfc6355>.

7.2. Informative References

[EANUCCGS]
EAN International and the Uniform Code Council, , "General EAN.UCC Specifications Version 5.0", Jan 2004.

[EPC-Tag-Data]
EPCglobal Inc., , "EPC(TM) Generation 1 Tag Data Standards Version 1.1 Rev.1.27 http://www.gs1.org/gsmp/kc/epcglobal/tds/tds_1_1_rev_1_27-standard-20050510.pdf", January 2005.

[IEEE802] IEEE, , "IEEE Std 802: IEEE Standards for Local and Metropolitan Networks: Overview and Architecture", 2001.

[IEEE802-eui]
IEEE, , "Guidelines for Use Organizationally Unique Identifier (OUI) and Company ID (CID) https://standards.ieee.org/develop/regauth/tut/eui.pdf", 2001.

[IEEE802-eui48]
IEEE, , "Guidelines for 48-Bit Global Identifier (EUI-48) https://standards.ieee.org/develop/regauth/tut/eui48.pdf", 2001.

[IEEE802-eui64]
IEEE, , "Guidelines for 64-Bit Global Identifier (EUI-64) https://standards.ieee.org/develop/regauth/tut/eui.pdf64", 2001.

[RFC3588] Calhoun, P., Loughney, J., Guttman, E., Zorn, G., and J. Arkko, "Diameter Base Protocol", RFC 3588, DOI 10.17487/RFC3588, September 2003, <http://www.rfc-editor.org/info/rfc3588>.

[RFID-DoD-spec]
Department of Defense, , "United States Department of Defense Suppliers Passive RFID Information Guide (Version 15.0)", January 2010.

[ThreeGPP-IDS]
3rd Generation Partnership Project, , "3GPP Technical Specification 23.003 V8.4.0: Technical Specification Group Core Network and Terminals; Numbering, addressing and identification (Release 8)", March 2009.

Authors’ Addresses

Charles E. Perkins
Futurewei Inc.
2330 Central Expressway
Santa Clara, CA 95050
USA

Phone: +1-408-330-4586
Email: charliep@computer.org

Vijay Devarapalli
Vasona Networks
2900 Lakeside Drive, Suite 180
Santa Clara, CA 95054
USA

DMM WG S. Gundavelli
Internet-Draft Cisco
Intended status: Informational S. Jeon
Expires: February 23, 2017 Sungkyunkwan University
August 22, 2016

DMM Deployment Models and Architectural Considerations
draft-ietf-dmm-deployment-models-00.txt

Abstract

This document identifies the deployment models for Distributed Mobility Management architecture.

Status of this Memo

Copyright Notice

Table of Contents

1. Overview

One of the key aspects of the Distributed Mobility Management (DMM) architecture is the separation of control plane (CP) and data plane (DP) functions of a network element. While data plane elements continue to reside on customized networking hardware, the control plane resides as a software element in the cloud. This is usually referred to as CP-DP separation and is the basis for the IETF’s DMM Architecture. This approach of centralized control plane and distributed data plane allows elastic scaling of control plane and efficient use of common data plane that is agnostic to access architectures.

This document identifies the functions in the DMM architecture and the supported deployment models.

2. Conventions and Terminology

2.1. Conventions

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].

2.2. Terminology

All the mobility related terms are to interpreted as defined in [RFC6275], [RFC5213], [RFC5844], [RFC7333], [RFC7429], [I-D.ietf-sfc-nsh] and [I-D.ietf-dmm-fpc-cpdp]. Additionally, this document uses the following terms:

Home Control-Plane Anchor (H-CPA)

The Home-CPA function hosts the mobile node’s mobility session. There can be more than one mobility session for a mobile node [MN] and those sessions may be anchored on the same or different Home-CPA’s. The home-CPA will interface with the home-dpa for managing the forwarding state.

Home Data Plane Anchor (Home-DPA)

The Home-DPA is the topological anchor for the mobile node’s IP address/prefix(es). The Home-DPA is chosen by the Home-CPA on a session-basis. The Home-DPA is in the forwarding path for all the mobile node’s IP traffic.

Access Control Plane Node (Access-CPN)

The Access-CPN is responsible for interfacing with the mobile node’s Home-CPA and with the Access-DPN. The Access-CPN has a protocol interface to the Home-CPA.

Access Data Plane Node (Access-DPN)

The Access-DPN function is hosted on the first-hop router where the mobile node is attached. This function is not hosted on a layer-2 bridging device such as a eNode(B) or Access Point.

3. DMM Architectural Overview

Following are the key goals of the Distributed Mobility Management architecture.

1. Separation of control and data Plane

2. Aggregation of control plane for elastic scaling

3. Distribution of the data plane for efficient network usage

4. Elimination of mobility state from the data plane

5. Dynamic selection of control and data plane nodes

6. Enabling the mobile node with network properties

7. Relocation of anchor functions for efficient network usage

3.1. DMM Service Primitives

The functions in the DMM architecture support a set of service primitives. Each of these service primitives identifies a specific service capability with the exact service definition. The functions in the DMM architecture are required to support a specific set of service primitives that are mandatory for that service function. Not all service primitives are applicable to all DMM functions. The below table identifies the service primitives that each of the DMM function SHOULD support. The marking "X" indicates the service primitive on that row needs to be supported by the identified DMM function on the corresponding column; for example, the IP address management must be supported by Home-CPA function.

Service Primitive	H-CPA	H-DPA	A-CPN	A-DPN	MC	RC
IP Management	X				X	
IP Anchoring		X				
MN Detect			X	X		
Routing		X		X		
Tunneling		X		X		
QoS Enforcement		X		X		
FPC Client	X		X		X	
FPC Agent		X		X		X
NSH Classifier		X		X		

Figure 1: Mapping of DMM functions

3.2. DMM Functions and Interfaces

3.2.1. Home Control-Plane Anchor (H-CPA):

The Home-CPA function hosts the mobile node’s mobility session. There can be more than one mobility session for a mobile node and those sessions may be anchored on the same or different Home-CPA’s. The home-CPA will interface with the homd-dpa for managing the forwarding state.

There can be more than one Home-CPA serving the same mobile node at a given point of time, each hosting a different control plane session.

The Home-CPA is responsible for life cycle management of the session, interfacing with the policy infrastructure, policy control and interfacing with the Home-DPA functions.

The Home-CPA function typically stays on the same node. In some special use-cases (Ex: Geo-Redundancy), the session may be migrated to a different node and with the new node assuming the Home-CPA role for that session.

3.2.2. Home Data-Plane Anchor (H-DPA):

The Home-DPA is the topological anchor for the mobile node’s IP address/prefix(es). The Home-DPA is chosen by the Home-CPA/MC on a session-basis. The Home-DPA is in the forwarding path for all the mobile node’s IP traffic.

As the mobile node roams in the mobile network, the mobile node’s access-DPN may change, however, the Home-DPA does not change, unless the session is migrated to a new node.

The Home-DPA interfaces with the Home-CPA/MC for all IP forwarding and QoS rules enforcement.

The Home-DPA and the Access-DPN functions may be collocated on the same node.

3.2.3. Access Control Plane Node (Access-CPN)

The Access-CPN is responsible for interfacing with the mobile node’s Home-CPA and with the Access-DPN. The Access-CPN has a protocol interface to the Home-CPA.

The Access-CPN is responsible for the mobile node’s Home-CPA selection based on: Mobile Node’s Attach Preferences, Access and Subscription Policy, Topological Proximity and Other Considerations.

The Access-CPN function is responsible for MN’s service authorization. It will interface with the access network authorization functions.

3.2.4. Access Data Plane Node (Access-DPN)

The Access-DPN function is hosted on the first-hop router where the mobile node is attached. This function is not hosted on a layer-2 bridging device such as a eNode(B) or Access Point.

The Access-DPA will have a protocol interface to the Access-CPA.

The Access-DPN and the Home-DPA functions may be collocated on the same node.

3.2.5. DMM Function Mapping to other Architectures

Following table identifies the potential mapping of DMM functions to protocol functions in other system architectures.

FUNCTION	PMIPv6	MIPv6	IPsec	3GPP	Broadband
Home-CPA	LMA-CPA	HA-CPA	IKE-CPA	PGW-CPA	BNG-CPA
Home-DPA	LMA-DPA	HA-DPA	IKE-DPA	PGW-DPA	BNG-DPA
Access-CPN	MAG-CPN	-	-	SGW-CPN	RG-CPN
Access-DPN	MAG-DPN	-	-	SGW-DPN	RG-DPN

Figure 2: Mapping of DMM functions

4. Deployment Models

This section identifies the key deployment models for the DMM architecture.

4.1. Model-1: Split Home Anchor Mode

In this model, the control and the data plane functions of the home anchor are separated and deployed on different nodes. The control plane function of the Home anchor is handled by the Home-CPA and where as the data plane function is handled by the Home-DPA. In this model, the access node operates in the legacy mode with the integrated control and user plane functions.

The FPC interface defined in [I-D.ietf-dmm-fpc-cpdp] allows the control plane functions to interact with the data plane for the subscriber’s forwarding state management.

```
                          +============+
                          |   Policy   |
             . . . . . . .|  Function  |. . . . . . .
             .            +============+            .
             .                                      .
             .                                      .
       +============+       {PMIPv6/GTP}      +============+
       |            |- - - - - - - - - - - - -|  Home-CPA  |
       |            |                         +============+
       |            |                               .
       |            |                               .  FPC
       | Access Node|                               .
       |            |                               .
       | (CPN + DPN)|                               .
       |            |                         +============+
       |   Legacy   |. . . . . . . . . . . . .|  Home-DPA  |
       +============+   UP {Tunnel/Route}     +============+
              .
              .
             +--+
             |MN|
             +--+
```

Figure 3: Split Home Anchor Mode

4.2. Model-2: Seperated Control and User Plane Mode

In this model, the control and the data plane functions on both the home anchor and the access node are seperated and deployed on different nodes. The control plane function of the Home anchor is handled by the Home-CPA and where as the data plane function is handled by the Home-DPA. The control plane function of the access node is handled by the Access-CPN and where as the data plane function is handled by the Access-DPN.

The FPC interface defined in [I-D.ietf-dmm-fpc-cpdp] allows the control plane functions of the home and access nodes to interact with the respective data plane functions for the subscriber’s forwarding state management.

```
                          +============+
                          |   Policy   |
             . . . . . . .|  Function  |. . . . . . .
             .            +============+            .
             .                                      .
             .                                      .
             .                                      .
             .                                      .
       +============+    {PMIPv6/GTP}        +============+
       | Access-CPN |- - - - - - - - - - - - |  Home-CPA  |
       +============+                        +============+
             .                                      .
             .  FPC                                 .  FPC
             .                                      .
             .                                      .
             .                                      .
       +============+                        +============+
       | Access-DPN |. . . . . . . . . . .   |  Home-DPA  |
       +============+   UP {Tunnel/Route}    +============+
              .
              .
             [MN]
```

Figure 4: Seperated Control and User Plane Mode

4.3. Model-3: Centralized Control Plane Mode

In this model, the control-plane functions of the home and the access nodes are collapsed. This is a flat architecture with no signaling protocol between the access node and home anchors. The interface between the Home-CPA and the Access-DPN is internal to the system.

The FPC interface defined in [I-D.ietf-dmm-fpc-cpdp] allows the mobility controller to interact with the respective data plane functions for the subscriber’s forwarding state management.

```
                    +=======================+     +=============+
                    | Home-CPA + Access-CPN |     |  Policy     |
                    |                       |-----| Function    |
                    +=======================+     +=============+
                               .
                            .     .
                         .          .
                   FPC .              .  FPC
                     .                   .
                   .                       .
          +============+                    +============+
          | Access-DPN |. . . . . .  . . . .|  Home-DPA  |
          +============+ UP {Tunnel/Route}  +============+
                .
                .
               [MN]
```

Figure 5: Centralized Control Plane Mode

4.4. Model-4: Data Plane Abstraction Mode

In this model, the data plane network is completely abstracted from the control plane. There is a new network element, Routing Controller which abstracts the entire data plane network and offers data plane services to the control plane functions. The control plane functions, Home-CPA and the Access-CPN interface with the Routing Controller for the forwarding state management.

The FPC interface defined in [I-D.ietf-dmm-fpc-cpdp] allows the Home-CPA and Access-CPN functions to interface with the Routing Controller for subscriber’s forwarding state management.

```
                          +============+
                          |   Policy   |
             . . . . . . .|  Function  |. . . . . . .
             .            +============+            .
             .                                      .
             .                                      .
             .                                      .
       +============+    {PMIPv6/GTP}        +============+
       | Access-CPN |- - - - - - - - - - - - |  Home-CPA  |
       +============+                        +============+
             .                                      .
             .                                      .
             .                                      .
             .           +============+             .
             . . . . . . |  Routing   | . . . . . . .
                         | Controller |
                         +============+
                                .
                             .     .
                          .          .  BGP/Others
                        .              .
                      .                   .
                    .                       .
          +============+                    +============+
          | Access-DPN |. . . . . .  . . . .|  Home-DPA  |
          +============+ UP {Tunnel/Route}  +============+
                .
                .
               [MN]
```

Figure 6: Data Plane Abstraction Mode

4.5. On-Demand Control Plane Orchestration Mode

In this model, there is a new function Mobility Controller which manages the orchestration of Access-CPN and Home-CPA functions. The Mobility Controller allocates the Home-CPA and Access-DPN

```
   + - - - - - - - - - - - - - - - - - - - - - - - - - - -+
   |    +----------+     +----------+     +----------+    |
        |Access-CPN|     |Access-CPN|     |Access-CPN|
   |    +----------+     +----------+     +----------+    |

   |    +----------+     +----------+     +----------+    |
        | Home-CPA |     | Home-CPA |     | Home-CPA |
   |    +----------+     +----------+     +----------+    |
   + - - - - - - - - - - - - - - - - - - - - - - - - - - -+
              .                 .
              .                 .
              .                 .
              .          +============+     +============+
              .          |   Mobility |     |  Policy    |
              .          | Controller |-----| Function   |
              .          +============+     +============+
              .
              .
              .
              .          +============+
              . . . . . .|  Routing   |
                         | Controller |
                         +============+
                                .
                                .
                                .
   + - - - - - - - - - - - - - - - - - - - - - - - - - - -+
   |    +----------+     +----------+     +----------+    |
        |Access-DPN|     |Access-DPN|     |Access-DPN|
   |    +----------+     +----------+     +----------+    |

   |    +----------+     +----------+     +----------+    |
        | Home-DPA |     | Home-DPA |     | Home-DPA |
   |    +----------+     +----------+     +----------+    |
   + - - - - - - - - - - - - - - - - - - - - - - - - - - -+
```

Figure 7: On-Demand CP Orchestration Mode

5. IANA Considerations

This document does not require any IANA actions.

6. Security Considerations

The control-plane messages exchanged between a Home-CPA and the Home-DPA must be protected using end-to-end security associations with data-integrity and data-origination capabilities.

IPsec ESP in transport mode with mandatory integrity protection should be used for protecting the signaling messages. IKEv2 should be used to set up security associations between the Home-CPA and Home-DPA.

There are no additional security considerations other than what is presented in the document.

7. Work Team

This document reflects contributions from the following work team members:

Younghan Kim

younghak@ssu.ac.kr

Vic Liu

liuzhiheng@chinamobile.com

Danny S Moses

danny.moses@intel.com

Marco Liebsch

liebsch@neclab.eu

Carlos Jesus Bernardos Cano

cjbc@it.uc3m.es

8. Acknowledgements

This document is a result of DMM WT#4 team discussions and ideas taken from several DMM WG presentations and documents including, draft-sijeon-dmm-deployment-models, draft-liu-dmm-deployment-scenario and others. The work teams would like to thank the authors of these documents and additionally the discussions in DMM Working group that

helped shape this document.

9. References

9.1. 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, <http://www.rfc-editor.org/info/rfc2119>.

9.2. Informative References

[I-D.ietf-dmm-fpc-cpdp] Liebsch, M., Matsushima, S., Gundavelli, S., Moses, D., and L. Bertz, "Protocol for Forwarding Policy Configuration (FPC) in DMM", draft-ietf-dmm-fpc-cpdp-03 (work in progress), March 2016.

[I-D.ietf-sfc-nsh] Quinn, P. and U. Elzur, "Network Service Header", draft-ietf-sfc-nsh-05 (work in progress), May 2016.

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

[RFC5844] Wakikawa, R. and S. Gundavelli, "IPv4 Support for Proxy Mobile IPv6", RFC 5844, DOI 10.17487/RFC5844, May 2010, <http://www.rfc-editor.org/info/rfc5844>.

[RFC6275] Perkins, C., Ed., Johnson, D., and J. Arkko, "Mobility Support in IPv6", RFC 6275, DOI 10.17487/RFC6275, July 2011, <http://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, <http://www.rfc-editor.org/info/rfc7333>.

[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, <http://www.rfc-editor.org/info/rfc7429>.

Authors' Addresses

Sri Gundavelli
Cisco
170 West Tasman Drive
San Jose, CA 95134
USA

Email: sgundave@cisco.com

Seil Jeon
Sungkyunkwan University
2066 Seobu-ro, Jangan-gu
Suwon, Gyeonggi-do
Korea

Email: seiljeon@skku.edu

DMM
Internet-Draft
Intended status: Informational
Expires: March 27, 2017

H. Chan, Ed.
X. Wei
Huawei Technologies
J. Lee
Sangmyung University
S. Jeon
Sungkyunkwan University
A. Petrescu
CEA, LIST
F. Templin
Boeing Research and Technology

September 23, 2016

Distributed Mobility Anchoring
draft-ietf-dmm-distributed-mobility-anchoring-02

Abstract

This document defines distributed mobility anchoring to meet diverse mobility needs in 5G Wireless and beyond. Multiple anchors and nodes with mobility functions work together to provide IP mobility support. A network or network slice may be configured with distributed mobility anchoring depending on the needs of mobility support. In the distributed mobility anchoring environment, multiple anchors are available for mid-session switching of an IP prefix anchor. Without an ongoing session, i.e., no IP session continuity required, a flow of a mobile node can be re-started using a new IP prefix which is allocated from a new network of the mobile node and is therefore anchored to the new network. With an ongoing session, the anchoring of the prior IP prefix may be relocated to the new network to enable IP session continuity.

Status of This Memo

This Internet-Draft will expire on March 27, 2017.

Copyright Notice

Table of Contents

1. Introduction

A key requirement in distributed mobility management [RFC7333] is to enable traffic to avoid traversing a single mobility anchor far from an optimal route. Distributed mobility management solutions do not make use of centrally deployed mobility anchor for a data plane [Paper-Distributed.Mobility]. As such, the traffic of a flow SHOULD be able to change from traversing one mobility anchor to traversing another mobility anchor as a mobile node (MN) moves, or when changing operation and management requirements call for mobility anchor switching, thus avoiding non-optimal routes. This draft proposes distributed mobility anchoring to enable making such route changes.

Distributed mobility anchoring employs multiple anchors in the data plane. In general, control plane functions may be separate from data plane functions and be centralized but may also be co-located with the data plane functions at the distributed anchors. Different configurations of distributed mobility anchoring are described in Section 3.1. For instance, the configurations for network-based mobility support in a flat network, for network-based mobility support in a hierarchical network, for host-based mobility support, and for NEtwork MObility (NEMO) basic support are described respectively in Section 3.1.1, Section 3.1.2, Section 3.1.3 and Section 3.1.4. Required operations and parameters for distributed mobility anchoring are presented in Section 3.2. For instance, location management is described in Section 3.2.1, forwarding management is described in Section 3.2.2.

An MN attached to an access router of a network or network slice may be allocated an IP prefix which is anchored to that router. It may then use an IP address configured from this prefix as the source IP address to run a flow with its correspondent node (CN). When there are multiple mobility anchors, an address selection for a given flow is first required before the flow is initiated. Using an anchor in an MN's network of attachment has the advantage that the packets can simply be forwarded according to the forwarding table. Although the anchor is in the MN's network of attachment when the flow was initiated, the MN may later move to another network, so that the IP no longer belongs to the current network of attachment of the MN.

Whether the flow needs IP session continuity will determine how to ensure that the IP address of the flow will be anchored to the new network of attachment. If the ongoing IP flow can cope with an IP prefix/address change, the flow can be reinitiated with a new IP address anchored in the new network as shown in Section 4.1. On the other hand, if the ongoing IP flow cannot cope with such change, mobility support is needed as shown in Section 4.2. A network or network slice supporting a mix of flows requiring and not requiring IP mobility support will need to distinguish these flows. The guidelines for such network or network slice are described in Section 4.1.1. The general guidelines for such network or network slice to provide IP mobility support are described in Section 4.2.1.

Specifically, IP mobility support can be provided by changing the anchoring of the IP prefix/address of the flow from the home network of the flow to the new network of attachment. The basic case may be with network-based mobility for a flat network configuration described in Section 5.1 with the guidelines described in Section 5.1.1. This case is discussed further with a centralized control plane in Section 5.2 with additional guidelines described in Section 5.2.1. A level of hierarchy of nodes may then be added to the network configuration. Mobility involving change in the Data Plane Node (DPN) without changing the Data Plane Anchor (DPA) is described in Section 5.3 with additional guidelines described in Section 5.3.1 Mobility involving change in the DPN without changing the DPA is described in Section 5.4 with additional guidelines described in Section 5.4.1

2. Conventions and Terminology

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

All general mobility-related terms and their acronyms used in this document are to be interpreted as defined in the Mobile IPv6 (MIPv6) base specification [RFC6275], the Proxy Mobile IPv6 (PMIPv6) specification [RFC5213], the "Mobility Related Terminologies" [RFC3753], and the DMM current practices and gap analysis [RFC7429]. These include terms such as mobile node (MN), correspondent node

(CN), home agent (HA), home address (HoA), care-of-address (CoA), local mobility anchor (LMA), and mobile access gateway (MAG).

In addition, this document uses the following terms:

Home network of an application session or a home address: the network that has allocated the HoA used for the session identifier by the application running in an MN. The MN may be running multiple application sessions, and each of these sessions can have a different home network.

IP prefix/address anchoring: An IP prefix, i.e., Home Network Prefix (HNP), or address, i.e., HoA, allocated to an MN is topologically anchored to an anchor node when the anchor node is able to advertise a connected route into the routing infrastructure for the allocated IP prefix.

Location Management (LM) function: managing and keeping track of the internetwork location of an MN. The location information may be a binding of the IP advertised address/prefix, e.g., HoA or HNP, to the IP routing address of the MN or of a node that can forward packets destined to the MN.

When the MN is a mobile router (MR) carrying a mobile network of mobile network nodes (MNN), the location information will also include the mobile network prefix (MNP), which is the IP prefix delegated to the MR. The MNP is allocated to the MNNs in the mobile network.

LM is a control plane function.

In a client-server protocol model, location query and update messages may be exchanged between a Location Management client (LMc) and a Location Management server (LMs).

Optionally, there may be a Location Management proxy (LMp) between LMc and LMs.

With separation of control plane and data plane, the LM function is in the control plane. It may be a logical function at the control plane node, control plane anchor, or mobility controller.

It may be distributed or centralized.

Forwarding Management (FM) function: packet interception and forwarding to/from the IP address/prefix assigned to the MN, based on the internetwork location information, either to the destination or to some other network element that knows how to forward the packets to their destination.

This function may be used to achieve traffic indirection. With separation of control plane and data plane, the FM function may split into a FM function in the data plane (FM-DP) and a FM function in the control plane (FM-CP).

FM-DP may be distributed with distributed mobility management. It may be a function in a data plane anchor or data plane node.

FM-CP may be distributed or centralized. It may be a function in a control plane node, control plane anchor or mobility controller.

Security Management (SM) function: The security management function controls security mechanisms/protocols providing access control, integrity, authentication, authorization, confidentiality, etc. for the control plane and data plane.

This function resides in all nodes such as control plane anchor, data plane anchor, mobile node, and correspondent node.

3. Distributed Mobility Anchoring

3.1. Configurations for Different Networks or Network Slices

The mobility functions may be implemented in different configurations of distributed mobility anchoring in architectures separating the control and data planes. The separation described in [I-D.ietf-dmm-deployment-models] has defined the home control plane anchor (Home-CPA), home data plane anchor (Home-DPA), access control plane node (Access-CPN), and access data plane node (Access-DPN), which are respectively abbreviated as CPA, DPA, CPN, and DPN here. Some configurations are described in [I-D.sijeon-dmm-deployment-models].

Different networks or different network slices may have different configurations in distributed mobility anchoring.

The configurations also differ depending on the desired mobility supports: network-based mobility support for a flat network in Section 3.1.1, network-based mobility support for a hierarchical network in Section 3.1.2, host-based mobility support

(Section 3.1.3), and NEtwork MObility (NEMO) based support in Section 3.1.4.

3.1.1. Network-based Mobility Support for a Flat Network

Figure 1 shows two different configurations of network-based mobility management for a flat network.

```
       (a)                                   (b)
                                              +-----+
                                              |LMs  |
                                              +-----+

 +------------+                        +------------+
 |CPA:        |                        |CPA:        |
 |FM-CP, LM   |                        |FM-CP, LMc  |
 +------------+                        +------------+
 +------------+ +------------+         +------------+ +------------+
 |DPA(IPa1):  | |DPA(IPa2):  |         |DPA(IPa1):  | |DPA(IPa2):  |
 |anchors IP1 | |anchors IP2 | ...     |anchors IP1 | |anchors IP2 | ...
 |FM-DP       | |FM-DP       |         |FM-DP       | |FM-DP       |
 +------------+ +------------+         +------------+ +------------+


 +------------+                        +------------+
 |MN(IP1)     |                        |MN(IP1)     |
 |flow(IP1,..)|                        |flow(IP1,..)|
 +------------+                        +------------+
```

Figure 1. Configurations of network-based mobility management for a flat network (a) FM-CP and LM at CPA, FM-DP at DPA; (b) Separate LMs, FM-CP and LMc at CPA, FM-DP at DPA.

Figure 1 also shows a distributed mobility anchoring environment with multiple instances of the DPA.

There is an FM-DP function at each of the distributed DPA.

The control plane may either be distributed (not shown) or centralized. When the CPA co-locates with the distributed DPA there will be multiple instances of the co-located CPA and DPA (not shown).

There is an FM-CP function at the CPA.

An MN is allocated an IP prefix/address IP1 which is anchored to the DPA with the IP prefix/address IPa1. The MN uses IP1 to communicate with a CN not shown in the figure. The flow of this communication session is shown as flow(IP1, ...) which uses IP1 and other parameters.

In Figure 1(a), LM and FM-CP co-locate at CPA.

Then LM may be distributed or centralized according to whether the CPA is distributed (not shown) or centralized.

Figure 1(b) differs from Figure 1(a) in that the LM function is split into a server LMs and a client LMc.

LMc and FM-CP co-locate at the CPA.

The LMs may be centralized whereas the LMc may be distributed or centralized according to whether the CPA is distributed (not shown) or centralized.

3.1.2. Network-based Mobility Support for a Hierarchical Network

Figure 2 shows two different configurations of network-based mobility management for a hierarchical network.

```
       (a)
 +------------+
 |CPA:        |
 |FM-CP, LMs  |
 +------------+
 +------------+                        +------------+
 |DPA(IPa1):  |                        |DPA(IPa2):  |
 |anchors IP1 |                        |anchors IP2 |                ...
 |FM-DP       |                        |FM-DP       |
 +------------+                        +------------+


 +------------+
 |CPN:        |
 |FM-CP, LMc  |
 +------------+
 +------------+ +------------+         +------------+ +------------+
 |DPN(IPn11): | |DPN(IPn12): |         |DPN(IPn21): | |DPN(IPn22)  |
 |FM-DP       | |FM-DP       | ...     |FM-DP       | |FM-DP       | ...
 +------------+ +------------+         +------------+ +------------+


 +------------+                        +------------+
 |MN(IP1)     |                        |MN(IP2)     |
 |flow(IP1,..)|                        |flow(IP2,..)|
 +------------+                        +------------+
```

Figure 2(a). Configurations of network-based mobility management for a hierarchical network with FM-CP and LMs at CPA, FM-DP at DPA; FM-CP and LMc at CPN, FM-DP at DPN.

```
       (b)
        +-----+
        |LMs  |
        +-----+

 +------------+
 |CPA:        |
 |FM-CP, LMp  |
 +------------+
 +------------+                        +------------+
 |DPA(IPa1):  |                        |DPA(IPa2):  |
 |anchors IP1 |                        |anchors IP2 |                ...
 |FM-DP       |                        |FM-DP       |
 +------------+                        +------------+


 +------------+
 |CPN:        |
 |FM-CP, LMc  |
 +------------+
 +------------+ +------------+         +------------+ +------------+
 |DPN(IPn11): | |DPN(IPn12): |         |DPN(IPn21): | |DPN(IPn22)  |
 |FM-DP       | |FM-DP       | ...     |FM-DP       | |FM-DP       | ...
 +------------+ +------------+         +------------+ +------------+


 +------------+                        +------------+
 |MN(IP1)     |                        |MN(IP2)     |
 |flow(IP1,..)|                        |flow(IP2,..)|
 +------------+                        +------------+
```

Figure 2(b). Configurations of network-based mobility management for a hierarchical network with separate LMs, FM-CP and LMp at CPA, FM-DP at DPA; FM-CP and LMc at CPN, FM-DP at DPN.

Figures 2 also shows a distributed mobility anchoring environment with multiple instances of the DPA.

In the hierarchy, there may be multiple DPN’s for each DPA.

There is FM-DP at each of the distributed DPA and at each of the distributed DPN.

The control plane may either be distributed (not shown) or centralized.

When the CPA co-locates with the distributed DPA there will be multiple instances of the co-located CPA and DPA (not shown).

When the CPN co-locates with the distributed DPN there will be multiple instances of the co-located CPN and DPN (not shown).

There is FM-CP function at the CPA and at the CPN.

MN is allocated an IP prefix/address IP1 which is anchored to the DPA with the IP prefix/address IPa1. It is using IP1 to communicate with a correspondent node (CN) not shown in the figure. The flow of this communication session is shown as flow(IP1, ...) which uses IP1 and other parameters.

In Figure 2(a), LMs and FM-CP are at the CPA. In addition, there are FM-CP and LMc at the CPN.

LMs may be distributed or centralized according to whether the CPA is distributed or centralized. The CPA may co-locate with DPA or may separate.

Figure 2(b) differs from Figure 2(a) in that the LMs is separated out, and a proxy LMp is added between the LMs and LMc.

LMp and FM-CP co-locate at the CPA.

FM-CP and LMc co-locate at the CPN.

The LMs may be centralized whereas the LMp may be distributed or centralized according to whether the CPA is distributed or centralized.

3.1.3. Host-based Mobility Support

Host-based variants of the mobility function configurations from Figures 2(a) and 2(b) are respectively shown in Figures 3(a) and 3(b) where the role to perform mobility functions by CPN and DPN are now taken by the MN. The MN then needs to possess the mobility functions FM and LMc.

```
       (a)                                    (b)
                                               +-----+
                                               |LMs  |
                                               +-----+

 +------------+                        +------------+
 |CPA:        |                        |CPA:        |
 |FM-CP, LMs  |                        |FM-CP, LMp  |
 +------------+                        +------------+
 +------------+ +------------+         +------------+ +------------+
 |DPA(IPa1):  | |DPA(IPa2):  |         |DPA(IPa1):  | |DPA(IPa2):  |
 |anchors IP1 | |anchors IP2 |  ...    |anchors IP1 | |anchors IP2 | ...
 |FM-DP       | |FM-DP       |         |FM-DP       | |FM-DP       |
 +------------+ +------------+         +------------+ +------------+


 +------------+                        +------------+
 |MN(IP1)     |                        |MN(IP1)     |
 |flow(IP1,..)|                        |flow(IP1,..)|
 |FM,    LMc  |                        |FM,    LMc  |
 +------------+                        +------------+
```

Figure 3. Configurations of host-based mobility management (a) FM-CP and LMs at CPA, FM-DP at DPA, FM and LMc at MN; (b) Separate LMs, FM-CP and LMp at CPA, FM-DP at DPA, FM and LMc at MN.

Figure 3 shows 2 configurations of host-based mobility management with multiple instances of DPA for a distributed mobility anchoring environment.

There is an FM-DP function at each of the distributed DPA.

The control plane may either be distributed (not shown) or centralized.

When the CPA co-locates with the distributed DPA there will be multiple instances of the co-located CPA and DPA (not shown).

There is an FM-CP function at the CPA.

The MN possesses the mobility functions such as FM and LMc.

The MN is allocated an IP prefix/address IP1 which is anchored to the DPA with the IP prefix/address IPa1. It is using IP1 to communicate with a CN not shown in the figure. The flow of this communication session is shown as flow(IP1, ...) which uses IP1 and other parameters.

In Figure 3(a), LMs and FM-CP co-locate at the CPA.

The LMs may be distributed or centralized according to whether the CPA is distributed (not shown) or centralized.

Figure 3(b) differs from Figure 3(a) in that the LMs is separated out and the proxy LMp is added between the LMs and LMc.

LMp and FM-CP co-locate at the CPA.

The LMs may be centralized whereas the LMp may be distributed or centralized according to whether the CPA is distributed (not shown) or centralized.

3.1.4. NEtwork MObility (NEMO) Basic Support

Figure 4 shows two configurations of NEMO basic support for a mobile router.

```
       (a)                                   (b)
                                              +-----+
                                              |LMs  |
                                              +-----+

 +------------+                        +------------+
 |CPA:        |                        |CPA:        |
 |FM-CP, LMs  |                        |FM-CP, LMp  |
 +------------+                        +------------+
 +------------+ +------------+         +------------+ +------------+
 |DPA(IPa1):  | |DPA(IPa2):  |         |DPA(IPa1):  | |DPA(IPa2):  |
 |anchors IP1 | |anchors IP2 |         |anchors IP1 | |anchors IP2 |
 |DHCPv6-PD   | |DHCPv6-PD   |  ...    |DHCPv6-PD   | |DHCPv6-PD   | ...
 |        IPn1| |        IPn2|         |        IPn1| |        IPn2|
 |FM-DP       | |FM-DP       |         |FM-DP       | |FM-DP       |
 +------------+ +------------+         +------------+ +------------+


 +------------+                        +------------+
 |FM-CP  LMc  |                        |FM-CP  LMc  |
 |- - - - - - |                        |- - - - - - |
 |MR(IP1)     |                        |MR(IP1)     |
 |anchors IPn1|                        |anchors IPn1|
 |FM-DP       |                        |FM-DP       |
 +------------+                        +------------+


 +------------+                        +------------+
 |MNN(IPn1)   |                        |MR(IP1n1)   |
 |flow(IPn1,.)|                        |flow(IPn1,.)|
 +------------+                        +------------+
```

Figure 4. Configurations of NEMO basic support for a MR. (a) FM-CP and LMs at CPA, FM-DP at DPA, FM and LMc at MR; (b) Separate LMs, FM-CP and LMp at CPA, FM-DP at DPA, FM and LMc at MR.

Figure 4 shows 2 configurations of host-based mobility management for a MR with multiple instances of DPA for a distributed mobility anchoring environment.

There is an FM-DP function at each of the distributed DPA.

The control plane may either be distributed (not shown) or centralized.

When the CPA co-locates with the distributed DPA there will be multiple instances of the co-located CPA and DPA (not shown).

There is FM-CP function at the CPA.

The MR possesses the mobility functions FM and LMc.

MR is allocated an IP prefix/address IP1 which is anchored to the DPA with the IP prefix/address IPa1.

A mobile network node (MNN) in the mobile network is allocated an IP prefix/address IPn1 which is anchored to the MR with the IP prefix/address IP1.

The MNN is using IPn1 to communicate with a correspondent node (CN) not shown in the figure. The flow of this communication session is shown as flow(IPn1, ...) which uses IPn1 and other parameters.

In Figure 4(a), LMs and FM-CP co-locate at the CPA.

The LMs may be distributed or centralized according to whether the CPA is distributed (not shown) or centralized.

Figure 4(b) differs from Figure 4(a) in that the LMs is separated out and the proxy LMp is added between the LMs and LMc.

LMp and FM-CP co-locate at the CPA.

The LMs may be centralized whereas the LMp may be distributed or centralized according to whether the CPA is distributed (not shown) or centralized.

3.2. Operations and Parameters

The operations of distributed mobility anchoring are defined in order that they may work together in expected manners to produce a distributed mobility solution. The needed information is passed as mobility message parameters, which must be protected in terms of integrity. Some parameters may require a means to support privacy of an MN or MR.

The mobility needs in 5G Wireless and beyond are diverse. Therefore operations needed to enable different distributed mobility solutions in different distributed mobility anchoring configurations are extensive as illustrated below. It is however not necessary for every distributed mobility solution to exhibit all the operations listed in this section. A given distributed mobility solution may exhibit the operations as needed.

3.2.1. Location Management

An example LM design consists of a distributed database with multiple LMs servers. The location information about the prefix/address of an MN is primarily at a given LMs. Peer LMs may exchange the location information with each other. LMc may retrieve a given record or send a given record update to LMs.

Location management configurations:

LM-cfg: As shown in Section 3.1:

LMs may be implemented at CPA, may co-locate with LMc at CPA, or may be a separate server.

LMc may be at CPA, CPN, or MN.

LMp may proxy between LMs and LMc.

Specifically:

Location management operations and parameters:

LM-cfg:1 LMs may co-locate with LMc at CPA in a flat network with network-based mobility as shown in Figure 1(a) in Section 3.1.1.

LM-cfg:2 LMs may be a separate server whereas LMc is implemented in CPA in a flat network with network-based mobility as shown in Figure 1(b) in Section 3.1.1.

LM-cfg:3 LMs may be implemented at CPA, whereas LMc is implemented at CPN in a hierarchical network with network-based mobility as shown in Figure 2(a) in Section 3.1.2 or at MN for host-based mobility as shown in Figure 3(a) in Section 3.1.3.

LM-cfg:4 LMs may be a separate server with LMp implemented at CPA whereas LMc is implemented at CPN in a hierarchical network with network-based mobility as shown in Figure 2(b) in Section 3.1.2 or at MN for host-based mobility as shown in Figure 3(b) in Section 3.1.3.

LM-db: LM may manage the location information in a client-server database system.

Example LM database functions are as follows:

LM-db:1 LMc may query LMs about location information for a prefix of MN (pull).
Parameters:
- IP prefix of MN: integrity support required and privacy support may be required.

LM-db:2 LMs may reply to LMc query about location information for a prefix of MN (pull).
Parameters:
- IP prefix of MN: integrity support required and privacy support may be required
- IP address of FM-DP/DPA/DPN to forward the packets of the flow: integrity support required.

LM-db:3 LMs may inform LMc about location information for a prefix of MN (push).
Parameters:
- IP prefix of MN: integrity support required and privacy support may be required
- IP address of FM-DP/DPA/DPN to forward the packets of the flow.

This function in the PMIPv6 protocol is the Update Notification (UPN) together with the Update Notification Acknowledgment (UPA) as defined in [RFC7077].

LM-db:4 LMc may inform LMs about update location information for a prefix of MN.
Parameters:
- IP prefix of MN: integrity support required and privacy support may be required
- IP address of FM-DP/DPA/DPN to forward the packets of the flow: integrity support required

This function in the MIPv6 / PMIPv6 protocol is the Binding Update (BU) / Proxy Binding Update (PBU) together with the Binding Acknowledgment (BA) / Proxy Binding Acknowledgment (PBA) as defined in [RFC6275] / [RFC5213] respectively.

LM-db:5 The MN may be a host or a router. When the MN is an MR, the prefix information may include the MNP delegated to the MR.
Additional parameters:
MNP: integrity support required and privacy support may be required

LM-svr: The LM may be a distributed database with multiple LMs servers.

For example:

LM-svr:1 A LMs may join a pool of LMs servers.
Parameters:
- IP address of the LMs: integrity support required
- IP prefixes for which the LMs will host the primary location information: integrity support required.

LM-svr:2 LMs may query a peer LMs about location information for a prefix of MN.
Parameters:
- IP prefix: integrity support required and privacy support may be required.

LM-svr:3 LMs may reply to a peer LMs about location information for a prefix of MN.
Parameters:
- IP prefix of MN: integrity support required and privacy support may be required
- IP address of FM-DP/DPA/DPN to forward the packets of the flow: integrity support required.

The parameters indicated above are only the minimal. In a specific mobility protocol, additional parameters should be added as needed. Examples of these additional parameters are those passed in the mobility options of the mobility header for MIPv6 [RFC6275] and for PMIPv6 [RFC5213].

3.2.2. Forwarding Management

Forwarding management configurations:

FM-cfg: As shown in Section 3.1:

FM-CP may be implemented at CPA, CPN, MN depending on the configuration chosen.

FM-DP may also be implemented at CPA, CPN, MN depending on the configuration chosen.

Specifically:

FM-cfg:1 FM-CP and FM-DP may be implemented at CPA and DPA respectively in a flat network with network-based mobility as shown in Figure 1(a) and Figure 1(b) in Section 3.1.1.

FM-cfg:2 FM-CP may be implemented at both CPA and CPN and FM-DP is implemented at both DPA and DPN in a hierarchical network with network-based mobility as shown in Figure 2(a) and Figure 2(b) in Section 3.1.2.

FM-cfg:3 FM-CP and FM-DP may be implemented at CPA and DPA respectively and also both implemented at MN for host-based mobility as shown in Figure 3(a) and Figure 3(b) in Section 3.1.3.

Forwarding management operations and parameters:

FM-find:1 An anchor may discover and be discovered such as through an anchor registration system as follows:

FM-find:2 FM registers and authenticates itself with a centralized mobility controller.
Parameters:
- IP address of DPA and its CPA: integrity support required
- IP prefix anchored to the DPA: integrity support required

registration reply: acknowledge of registration and echo the input parameters.

FM-find:3 FM discovers the FM of another IP prefix by querying the mobility controller based on the IP prefix.
Parameters:
- IP prefix of MN: integrity support required and privacy support may be required

FM-find:4 when making anchor discovery FM expects the answer parameters:
- IP address of DPA to which IP prefix of MN is anchored: integrity support required
- IP prefix of the corresponding CPA: integrity support required

FM-flow:1 The FM may be carried out on the packets to/from an MN up to the granularity of a flow.

FM-flow:2 Example matching parameters are in the 5-tuple of a flow.

FM-cpdp: With separation of control plane function and data plane function, FM-CP and FM-DP communicate with each other. Such communication may be realized by the appropriate messages in [I-D.ietf-dmm-fpc-cpdp].

For example:

FM-cpdp:1 CPA/FM-CP sends forwarding table updates to DPA/FM-DP. Parameters:
- New forwarding table entries to add: integrity support required
- Expired forwarding table entries to delete: integrity support required

FM-cpdp:2 DPA/FM-DP sends to CPA/FM-CP about its status and load. Parameters:
- State of forwarding function being active or not: integrity support required
- Loading percentage: integrity support required

FM-path:1 FM may change the forwarding path of a flow upon a change of point of attachment of a MN. Prior to the changes, packets coming from the CN to the MN would traverse from the CN to the home network anchor of the flow for the MN before reaching the MN. Changes are from this original forwarding path or paths to a new forwarding path or paths from the CN to the current AR of the MN and then the MN itself.

FM-path:2 As an incoming packet is forwarded from the CN to the MN, the far end where forwarding path change begins may in general be any node in the original forwarding path from the CN to the home network DPA. The packet is forwarded to the MN for host-based mobility and to a node in the network which will deliver the packets to the MN for network-based mobility. The near-end is generally a DPN with a hierarchical network but may also be another node with DPA capability in a flattened network.

FM-path:3 The mechanisms to accomplish such changes may include changes to the forwarding table and indirection such as tunneling, rewriting packet header, or NAT.

Note: An emphasis in this document in distributed mobility anchoring is to explain the use of multiple anchors to avoid unnecessarily long route which may be encountered in centralized mobility anchoring. It is therefore not the emphasis of this document on which particular mechanism to choose from.

FM-path-tbl:4 With forwarding table updates, changes to the forwarding table are needed at each of the affected forwarding switches in order to change the forwarding path of the packets for the flow from that originally between the CN and the home network anchor to that between the CN and the new AR.

Forwarding table updates may be achieved through BGP update as described in [I-D.templin-aerolink], [I-D.mccann-dmm-flatarch] and also for 3GPP Evolved Packet Core (EPC) network in [I-D.matsushima-stateless-uplane-vepc] when the scope and response time can be managed. Alternatively, a centralized control plane may be used.

When the control plane is centralized, forwarding table updates may be achieved through messaging between the centralized control plane and the distributed forwarding switches as described above (FM-cpdp) in this section.

Forwarding table updates may be triggered using DHCPv6-PD prefix delegation to change the role of IP anchoring from the home network anchor (with FM-DP) to the new anchor (with FM-DP) to which the MN is currently attached. The new anchor will then advertise routes for the delegated prefix.

With a distributed routing protocol, the updates spread out from neighbors to neighbors and will affect all the forwarding switches such that the packets sent from "any" node to MN will go to the new AR.

Yet the scope of such updates for a given flow may be confined to only those forwarding switches such that the packets sent only from the "CN" to MN will go to the new AR. Such confinement may be made when using a centralized central plane possessing a global view of all the forwarding switches.

FM-path-tbl:5 FM reverts the changes previously made to the forwarding path of a flow when such changes are no longer needed, e.g., when all the ongoing flows using an IP prefix/address requiring IP session continuity have closed. When using DHCPv6-PD, the forwarding paths will be reverted upon prefix lease expiration.

FM-path-ind:6 Indirection forwards the incoming packets of the flow from the DPA at the far end to a DPA/DPN at the near end of indirection. Both ends of the indirection needs to know the LM information of the MN for the flow and also needs to possess FM capability to perform indirection.

FM-path-ind:7 The mechanism of changing the forwarding path in [RFC6275] and [RFC5213] is tunneling. In the control plane, the FM-CP sets up the tunnel by instructing the FM-DP at both ends of the tunnel. In the data plane, the FM-DP at the start of the tunnel performs packet encapsulation, whereas the FM-DP at the end of the tunnel decapsulates the packet.

Note that in principle the ends of the indirection path can be any pair of network elements with the FM-DP function.

FM-path-ind:8 FM reverts the changes previously made to the forwarding path of a flow when such changes are no longer needed, e.g., when all the ongoing flows using an IP prefix/address requiring IP session continuity have closed. When tunneling is used, the tunnels will be torn down when they are no longer needed.

FM-DPA:1 Recall from above that for the incoming packets from the CN, forwarding path change by FM is from the DPA at the far end which may be at any forwarding switch (or even CN itself) in the original forwarding path to the near end DPA/DPN.

It is necessary that any incoming packet from the CN of the flow must traverse the DPA (or at least one of the DPAs, e.g., in the case of anycast) at the far end in order for the packet to detour to a new forwarding path.

Therefore a convenient design is to locate the far end DPA at a unique location which is always in the forwarding path. This is the case in a centralized mobility design where the DPA at the far end is the home network anchor of the flow.

Distributed mobility however may place the far end DPA at other locations in order to avoid unnecessarily long route.

FM-DPA:2 With multiple nodes possessing DPA capabilities, the role of FM to begin path change for the incoming packets of a flow at the home network DPA at the far end may be passed to or added to that of another DPA.

In particular, this DPA role may be moved upstream from the home network DPA in the original forwarding path from CN to MN.

FM-DPA:3 Optimization of the new forwarding path may be achieved when the path change for the incoming packets begins at a DPA where the original path and the direct IPv6 path overlaps. Then the new forwarding path will resemble the direct IPv6 path from the CN to the MN.

FM-DPA-tbl:4 Forwarding table updates, such as that triggered using DHCPv6-PD to change the role of IP anchoring from the home network anchor (DPA with FM-DP) to the new anchor (DPA with FM-DP), may put the near end of the path change at the new DPA. Subsequent forwarding table updates may propagates upstream up to a far end where the original path and the direct IPv6 path overlaps.

When that far end is too far upstream the signaling of forwarding table updates may become excessive. An alternative is to use indirection (see FM-DPA-ind) from that far end to the new DPA at the near end.

Still another alternative is to combine forwarding table update with indirection.

FM-DPA-tbl:5 Changes made by FM to the following tables, which are IPv6 nodes, at the ends of the path change for a flow will be reverted when the mobility support for the flow is no longer needed, e.g., when the flows have terminated.

FM-DPA-ind:6 With indirection, locating or moving the FM function to begin indirection upstream along the forwarding path from CN to MN again may help to reduce unnecessarily long path.

FM-DPA-ind:7 Changes made by FM to establish indirection at the DPA and DPN, which are IPv6 nodes, at the ends of the path change for a flow will be reverted when the mobility support for the flow is no longer needed, e.g., when the flows have terminated.

FM-state:1 In addition to the above, a flow/session may contain states with the required information for QoS, charging, etc. as needed. These states need to be transferred from the old anchor to the new anchor.

```
   FM-buffer:1  An anchor can buffer packets of a flow in a mobility
                event:
   FM-buffer:2  CPA/FM-CP informs DPA/FM-DP to buffer packets of a flow.
                Trigger:
                - MN leaves DPA in a mobility event.
                Parameters:
                - IP prefix of the flow for which packets need to be
                buffered: integrity support required

   FM-buffer:3  CPA/FM-CP on behalf of a new DPA/FM-DP informs the CPA/
                FM-CP of the prior DPA/FM-DP that it is ready to receive
                any buffered packets of a flow.
                Parameters:
                - Destination IP prefix of the flow's packets: integrity
                support required
                - IP address of the new DPA: integrity support required
```

FM-mr:1 When the MN is a mobile router the access router anchoring the IP prefix of MR will also anchor the IP prefix or prefixes delegated to the MR.

4. IP Mobility Handling in Distributed Anchoring Environments - Mobility Support Only When Needed

IP Mobility Support Only When Needed:

IP mobility support may be provided only when needed instead of being provided by default. The LM and FM functions in the different configurations shown in Section 3.1 are then utilized only when needed.

A straightforward choice of mobility anchoring is for a flow to use the IP prefix of the network to which the MN is attached when the flow is initiated [I-D.seite-dmm-dma].

The IP prefix/address at the MN's side of a flow may be anchored at the access router to which the MN is attached. For example, when an

MN attaches to a network (Net1) or moves to a new network (Net2), it is allocated an IP prefix from the attached network. In addition to configuring new link-local addresses, the MN configures from this prefix an IP address which is typically a dynamic IP address. It then uses this IP address when a flow is initiated. Packets to the MN in this flow are simply forwarded according to the forwarding table.

There may be multiple IP prefixes/addresses that an MN can select when initiating a flow. They may be from the same access network or different access networks. The network may advertise these prefixes with cost options [I-D.mccann-dmm-prefixcost] so that the mobile node may choose the one with the least cost. In addition, these IP prefixes/addresses may be of different types regarding whether mobility support is needed [I-D.ietf-dmm-ondemand-mobility]. A flow will need to choose the appropriate one according to whether it needs IP mobility support.

4.1. No Need of IP Mobility: Changing to New IP Prefix/Address

When IP mobility support is not needed for a flow, the LM and FM functions are not utilized so that the configurations in Section 3.1 are simplified as shown in Figure 5.

```
Net1                                                   Net2

+---------------+                                      +---------------+
|AR1            |                                      |AR2            |
+---------------+                                      +---------------+
|CPA:           |                                      |CPA:           |
|---------------|                                      |---------------|
|DPA(IPa1):     |                                      |DPA(IPa2):     |
|anchors IP1    |                                      |anchors IP2    |
+---------------+                                      +---------------+


+...............+                                      +---------------+
.MN(IP1)        .                 move                 |MN(IP2)        |
.flow(IP1,...)  .               =======>               |flow(IP2,...)  |
+...............+                                      +---------------+
```

Figure 5. Changing to the new IP prefix/address. MN running a flow using IP1 in a network Net1 changes to running a flow using IP2 in Net2.

When there is no need to provide IP mobility to a flow, the flow may use a new IP address acquired from a new network as the MN moves to the new network.

Regardless of whether IP mobility is needed, if the flow has terminated before the MN moves to a new network, the flow may subsequently restart using the new IP address allocated from the new network.

When IP session continuity is needed, even if a flow is ongoing as the MN moves, it may still be desirable for the flow to change to using the new IP prefix configured in the new network. The flow may then close and then restart using a new IP address configured in the new network. Such a change in the IP address of the flow may be enabled using a higher layer mobility support which is not in the scope of this document.

In Figure 5, a flow initiated while the MN was in a network Net1 has terminated before the MN moves to a new network Net2. After moving to Net2, the MN uses the new IP prefix anchored in Net2 to start a new flow. The packets may then be forwarded without requiring IP layer mobility support.

An example call flow is outlined in Figure 6.

```
 MN                    p-AR          n-AR                             CN
  |MN attaches to p-AR: |             |                                |
  |acquire MN-ID and profile          |                                |
  |--RS---------------->|             |                                |
  |                     |             |                                |
  |<----------RA(HNP1)--|             |                                |
  |                     |             |                                |
Allocated prefix HNP1
IP1 address configuration
  |                     |             |                                |
  |<-Flow(IP1,IPcn,...)-+--------------------------------------------->|
  |                     |             |                                |
  |MN detaches from p-AR|             |                                |
  |MN attaches to n-AR  |             |                                |
  |                     |             |                                |
  |--RS------------------------------>|                                |
  |                     |             |                                |
  |<--------------RA(HNP2)------------|                                |
  |                     |             |                                |
Allocated prefix HNP2
IP2 address configuration
  |                     |             |                                |
  |<-new Flow(IP2,IPcn,...)-----------+------------------------------->|
  |                     |             |                                |
```

Figure 6. Re-starting a flow to use the IP allocated from the network at which the MN is attached.

4.1.1. Guidelines for IPv6 Nodes: Changing to New IP Prefix/Address

A network or network slice may not need IP mobility support. For example, a network slice for stationary sensors only will never encounter mobility.

The standard functions in IPv6 already include dropping the old IPv6 prefix/address and acquiring new IPv6 prefix/address when the node changes its point of attachment to a new network. Therefore, a network or network slice not providing IP mobility support at all will not need any of the functions with the mobility operations and messages described in Section 3.2.

The guidelines for the IPv6 nodes in a network or network slice supporting a mix of flows requiring and not requiring IP mobility support include the following:

GL-cfg:1 A network or network slice supporting a mix of flows requiring and not requiring mobility support may take any of the configurations described in Section 3.1 and need to implement in the appropriate IPv6 nodes the mobility functions LM and FM as described respectively in LM-cfg and FM-cfg in Section 3.2 according to the configuration chosen.

GL-mix:1 These mobility functions perform some of the operations with the appropriate messages as described in Section 3.2 depending on which mobility mechanisms are used. Yet these mobility functions must not be invoked for a flow that does not need IP mobility support. It is necessary to be able to distinguish the needs of a flow. The guidelines for the MN and the AR are in the following.

GL-mix:2 Regardless of whether there are flows requiring IP mobility support, when the MN changes its point of attachment to a new network, it needs to configure a new global IP address for use in the new network in addition to configuring the new link-local addresses.

GL-mix:3 The MN needs to check whether a flow needs IP mobility support. This can be performed when the application was initiated. The specific method is not in the scope of this document.

GL-mix:4 The information of whether a flow needs IP mobility support is conveyed to the network such as by choosing an IP address to be provided with mobility support as described in [I-D.ietf-dmm-ondemand-mobility]. Then as the MN attaches to a new network, if the MN was using an IP address that is not supposed to be provided with mobility support, the access router will not invoke the mobility functions described in Section 3.2 for this IP address. That is, the IP address from the prior network is simply not used in the new network.

The above guidelines are only to enable distinguishing whether there is need of IP mobility support for a flow that does not. When the flow needs IP mobility support, the list of guidelines will continue in Section 4.2.1.

4.2. Need of IP Mobility

When IP mobility is needed for a flow, the LM and FM functions in Section 3.1 are utilized. The mobility support may be provided by IP prefix anchor switching to the new network to be described in Section 5 or by using other mobility management methods

([Paper-Distributed.Mobility.PMIP] and [Paper-Distributed.Mobility.Review]). Then the flow may continue to use the IP prefix from the prior network of attachment. Yet some time later, the user application for the flow may be closed. If the application is started again, the new flow may not need to use the prior network's IP address to avoid having to invoke IP mobility support. This may be the case where a dynamic IP prefix/address rather than a permanent one is used. The flow may then use the new IP prefix in the network where the flow is being initiated. Routing is again kept simpler without employing IP mobility and will remain so as long as the MN which is now in the new network has not moved again and left to another new network.

An example call flow in this case is outlined in Figure 7.

```
 MN                    p-AR          n-AR                             CN
  |MN attaches to p-AR: |             |                                |
  |acquire MN-ID and profile          |                                |
  |--RS---------------->|             |                                |
  |                     |             |                                |
  |<----------RA(HNP1)--|             |                                |
  |                     |             |                                |
Allocated prefix HNP1
IP1 address configuration
  |                     |             |                                |
  |<-Flow(IP1,IPcn,...)-+--------------------------------------------->|
  |                     |             |                                |
  |MN detach from p-AR  |             |                                |
  |MN attach to n-AR    |             |                                |
  |                     |             |                                |
  |--RS------------------------------>|                                |

IP mobility support such as that described in next sub-section
  |<--------------RA(HNP2,HNP1)-------|                                |
  |                     |             |                                |
  |<-Flow(IP1,IPcn,...)---------------+------------------------------->|
  |                     |             |                                |
Allocated prefix HNP2
IP2 address configuration
  |                     |             |                                |
Flow(IP1,IPcn) terminates
  |                     |             |                                |
  |<-new Flow(IP2,IPcn,...)-----------+------------------------------->|
  |                     |             |                                |
```

Figure 7. A flow continues to use the IP from its home network after MN has moved to a new network.

4.2.1. Guidelines for IPv6 Nodes: Need of IP Mobility

The configuration guidelines of distributed mobility for the IPv6 nodes in a network or network slice supporting a mix of flows requiring and not requiring distributed mobility support are as follows:

GL-cfg:2 Multiple instances of DPAs (at access routers) which are providing IP prefix to the MNs are needed to provide distributed mobility anchoring in an appropriate configuration such as those in Figure 1 (Section 3.1.1) for network-based distributed mobility or in Figure 3 (Section 3.1.3) for host-based distributed mobility.

The appropriate IPv6 nodes (CPA, DPA, CPN, DPN) are to be implemented the mobility functions LM and FM as described respectively in LM-cfg and FM-cfg in Section 3.2 according to the configuration chosen.

The guidelines of distributed mobility for the IPv6 nodes in a network or network slice supporting a mix of flows requiring and not requiring distributed mobility support had begun with those given as GL-mix in Section 4.1.1 and continue as follows:

GL-mix:5 The distributed anchors may need to message with each other. When such messaging is needed, the anchors may need to discover each other as described in the FM operations and mobility message parameters (FM-find) in Section 3.2.2.

GL-mix:6 The anchors may need to provide mobility support on a per-flow basis as described in the FM operations and mobility message parameters (FM-flow) in Section 3.2.2.

GL-mix:7 Then the anchors need to properly forward the packets of the flows as described in the FM operations and mobility message parameters (FM-path, FM-path-tbl, FM-DPA, FM-DPA-tbl) in Section 3.2.2.

GL-mix:8 If there are in-flight packets toward the old anchor while the MN is moving to the new anchor, it may be necessary to buffer these packets and then forward to the new anchor after the old anchor knows that the new anchor is ready. Such are described in the FM operations and mobility message parameters (FM-buffer) in Section 3.2.2.

5. IP Mobility Handling in Distributed Mobility Anchoring Environments - Anchor Switching to the New Network

IP Prefix/Address Anchor Switching to the New Network:

IP mobility is invoked to enable IP session continuity for an ongoing flow as the MN moves to a new network. Here the anchoring of the IP address of the flow is in the home network of the flow, which is not in the current network of attachment. A centralized mobility management mechanism may employ indirection from the anchor in the home network to the current network of attachment. Yet it may be difficult to avoid unnecessarily long route when the route between the MN and the CN via the anchor in the home network is significantly longer than the direct route between them. An alternative is to switch the IP prefix/address anchoring to the new network.

5.1. IP Prefix/Address Anchor Switching for Flat Network

The IP prefix/address anchoring may move without changing the IP prefix/address of the flow. Here the LM and FM functions in Figures 1(a) and 1(b) in Section 3.1 are implemented as shown in Figure 8.

```
Net1                                                   Net2

+---------------+                                      +---------------+
|AR1            |                                      |AR2            |
+---------------+                                      +---------------+
|CPA:           |                                      |CPA:           |
|LM:IP1<-->IPa2 |                                      |LM:IP1<-->IPa2 |
|---------------|                                      |---------------|
|DPA(IPa1):     |                                      |DPA(IPa2):     |
|anchors IP1    |                 move                 |anchors IP2,IP1|
|FM:DHCPv6-PD   |               =======>               |FM:DHCPv6-PD   |
+---------------+                                      +---------------+


+...............+                                      +---------------+
.MN(IP1)        .                 move                 |MN(IP2,IP1)    |
.flow(IP1,...)  .               =======>               |flow(IP1,...)  |
+...............+                                      +---------------+
```

Figure 8. IP prefix/address anchor switching to the new network. MN with flow using IP1 in Net1 continues to run the flow using IP1 as it moves to Net2.

As an MN with an ongoing session moves to a new network, the flow may preserve IP session continuity by moving the anchoring of the original IP prefix/address of the flow to the new network. BGP

UPDATE messages may be used to change the forwarding table entries as described in [I-D.templin-aerolink] and [I-D.mccann-dmm-flatarch] if the response time of such updates does not exceed the handover delay requirement of the flow. An alternative is to use a centralized routing protocol to be described in Section 5.2 with a centralized control plane.

5.1.1. Guidelines for IPv6 Nodes: Switching Anchor for Flat Network

The configuration guideline for a flat network or network slice supporting a mix of flows requiring and not requiring IP mobility support is:

GL-cfg:3 Multiple instances of DPAs (at access routers) which are providing IP prefix to the MNs are needed to provide distributed mobility anchoring according to Figure 1(a) or Figure 1(b)in Section 3.1 for a flat network.

The appropriate IPv6 nodes (CPA, DPA) are to be implemented the mobility functions LM and FM as described respectively in LM-cfg:1 or LM-cfg:2 and FM-cfg:1 in Section 3.2.

The guidelines (GL-mix) in Section 4.1.1 and in Section 4.2.1 for the IPv6 nodes for a network or network slice supporting a mix of flows requiring and not requiring IP mobility support apply here. In addition, the following are required.

GL-switch:1 The location management provides information about which IP prefix from an AR in the original network is being used by a flow in which AR in a new network. Such information needs to be deleted or updated when such flows have closed so that the IP prefix is no longer used in a different network. The LM operations are described in Section 3.2.1.

GL-switch:2 The FM functions are implemented through the DHCPv6-PD protocol. Here the anchor operations to properly forward the packets for a flow as described in the FM operations and mobility message parameters in Section 3.2.2 FM-path, FM-path-tbl, FM-DPA, FM-DPA-tbl are realized by changing the anchor with DHCPv6-PD and also by reverting such changes later after the application has already closed and when the DHCPv6-PD timer expires. If there are in-flight packets toward the old anchor while the MN is moving to the new anchor, it may be necessary to buffer these packets and then forward to the new anchor after the old anchor knows that the new anchor is ready as are described in Section 3.2.2 (FM-buffer). The anchors may also need to discover each other as described also in the FM operations and mobility message parameters (FM-find).

GL-switch:3 The security management function in the anchor node at a new network must allow to assign the original IP prefix/ address used by the mobile node at the previous (original) network. As the assigned original IP prefix/ address is to be used in the new network, the security management function in the anchor node must allow to advertise the prefix of the original IP address and also allow the mobile node to send and receive data packets with the original IP address.

GL-switch:4 The security management function in the mobile node must allow to configure the original IP prefix/address used at the previous (original) network when the original IP prefix/address is assigned by the anchor node in the new network. The security management function in the mobile node also allows to use the original IP address for the previous flow in the new network.

5.2. IP Prefix/Address Anchor Switching for Flat Network with Centralized Control Plane

An example of IP prefix anchor switching is in the case where Net1 and Net2 both belong to the same operator network with separation of control and data planes ([I-D.liu-dmm-deployment-scenario] and [I-D.matsushima-stateless-uplane-vepc]), where the controller may send to the switches/routers the updated information of the forwarding tables with the IP address anchoring of the original IP prefix/address at AR1 moved to AR2 in the new network. That is, the IP address anchoring in the original network which was advertising the prefix will need to move to the new network. As the anchoring in the new network advertises the prefix of the original IP address in the new network, the forwarding tables will be updated so that packets of the flow will be forwarded according to the updated forwarding tables. The configurations in Figures 1(a) and 1(b) in Section 3.1 for which FM-CP and LM are centralized and FM-DP’s are distributed apply here. Figure 9 shows its implementation where LM is a binding between the original IP prefix/address of the flow and the IP address of the new DPA, whereas FM uses the DHCPv6-PD protocol.

```
Net1                                                   Net2
+----------------------------------------------------------------------+
|                            CPA:                                      |
|                            LM:IP1<-->IPa2                            |
|                            FM-CP                                     |
+----------------------------------------------------------------------+


+---------------+                                      +---------------+
|AR1            |                                      |AR2            |
+---------------+                                      +---------------+
|DPA(IPa1):     |                                      |DPA(IPa2):     |
|anchors IP1    |                 move                 |anchors IP2,IP1|
|FM:DHCPv6-PD   |               =======>               |FM:DHCPv6-PD   |
+---------------+                                      +---------------+


+...............+                                      +---------------+
.MN(IP1)        .                 move                 |MN(IP2,IP1)    |
.flow(IP1,...)  .               =======>               |flow(IP1,...)  |
+...............+                                      +---------------+
```

Figure 9. IP prefix/address anchor switching to the new network with with LM and FM-CP in a centralized control plane whereas the FM-DP's are distributed.

The example call flow in Figure 10 shows that MN is allocated HNP1 when it attaches to the p-AR. A flow running in MN and needing IP mobility may continue to use the previous IP prefix by moving the anchoring of the IP prefix to the new network. Yet a new flow to be initiated in the new network may simply use a new IP prefix allocated from the new network.

```
 MN                    p-AR          n-AR            DHCPv6 Servers   CN
  |MN attaches to p-AR: |             |                     |          |
  |acquire MN-ID and profile          |                     |          |
  |--RS---------------->|             |                     |          |
  |<----------RA(HNP1)--|             |                     |          |
  |                     |             |             Allocate MN-HNP1   |
IP addr config          |             |                     |          |
  |                     |             |                     |          |
  |<-Flow(IP1,IPcn,...)-+--------------------------------------------->|
  |                     |             |                     |          |
  |MN detach from p-AR  |             |                     |          |
  |MN attach to n-AR    |             |                     |          |
  |                     |             |                     |          |
  |--RS------------------------------>|                     |          |
  |                     |             |                     |          |
  |                     |------DHCPv6 release-------------->|          |
  |                     |             |                     |          |
  |                     |             |--DHCPv6 PD request->|          |
  |                     |             |<-DHCPv6 PD reply--->|          |
  |                     |             |                     |          |
  |                 forwarding table updates                |          |
  |                     |             |                     |          |
  |<--------------RA(HNP2,HNP1)-------|                     |          |
  |                     |             |             Allocate MN-HNP2   |
IP addr config          |             |                     |          |
  |                     |             |                     |          |
  |<-Flow(IP1,IPcn,...)---------------+------------------------------->|
  |                     |             |                     |          |
  |  Flow(IP1,IPcn,...) terminates    |                     |          |
  |                     |             |                     |          |
  |                     | DHCPv6-PD timeout                 |          |
  |                     |             |                     |          |
  |                 forwarding table updates                |          |
  |                     |             |                     |          |
  |                     |             |                     |          |
  |<-new Flow(IP2,IPcn,...)-----------+------------------------------->|
  |                     |             |                     |          |
```

Figure 10. DMM solution. MN with flow using IP1 in Net1 continues to run the flow using IP1 as it moves to Net2.

As the MN moves from p-AR to n-AR, the p-AR as a DHCPv6 client may send a DHCPv6 release message to release the HNP1. It is now necessary for n-AR to learn the IP prefix of the MN from the previous network so that it will be possible for Net2 to allocate both the previous network prefix and the new network prefix. The network may learn the previous prefix in different methods. For example, the MN may provide its previous network prefix information by including it to the RS message [I-D.jhlee-dmm-dnpp].

Knowing that MN is using HNP1, the n-AR sends to a DHCPv6 server a DHCPv6-PD request to move the HNP1 to n-AR. The server sends to n-AR a DHCPv6-PD reply to move the HNP1. Then forwarding tables updates will take place here.

In addition, the MN also needs a new HNP in the new network. The n-AR may now send RA to n-AR, with prefix information that includes HNP1 and HNP2. The MN may then continue to use IP1. In addition, the MN is allocated the prefix HNP2 with which it may configure its IP addresses. Now for flows using IP1, packets destined to IP1 will be forwarded to the MN via n-AR.

As such flows have terminated and DHCPv6-PD has timed out, HNP1 goes back to Net1. MN will then be left with HNP2 only, which it will use when it now starts a new flow.

5.2.1. Additional Guidelines for IPv6 Nodes: Switching Anchor with Centralized CP

The configuration guideline for a flat network or network slice with centralized control plane and supporting a mix of flows requiring and not requiring IP mobility support is:

GL-cfg:4 Multiple instances of DPAs (at access routers) which are providing IP prefix to the MNs are needed to provide distributed mobility anchoring according to Figure 1(a) or Figure 1(b)in Section 3.1 with centralized control plane for a flat network.

The appropriate IPv6 nodes (CPA, DPA) are to be implemented the mobility functions LM and FM as described respectively in LM-cfg:1 or LM-cfg:2 and FM-cfg:1 in Section 3.2.

The guidelines (GL-mix) in Section 4.1.1 and in Section 4.2.1 for the IPv6 nodes for a network or network slice supporting a mix of flows requiring and not requiring IP mobility support apply here. The guidelines (GL-mix) in Section 5.1.1 also apply here. In addition, the following are required.

GL-switch:5 The anchor operations to properly forward the packets for a flow as described in the FM operations and mobility message parameters in Section 3.2.2 FM-path, FM-path-tbl, FM-DPA, FM-DPA-tbl is realized by changing the anchoring with DHCPv6-PD and undoing such changes later when its timer expires and the application has already closed. With the anchors being separated in control and data planes with LMs and FM-CP centralized in the same control plane, messaging between anchors and the discovery of anchors become internal to the control plane as described in Section 3.2.2 FM-cpdp. However, the centralized FM-CP needs to communicate with the distributed FM-DP as described as described in the FM operations and mobility message parameters (FM-find). Such may be realized by the appropriate messages in [I-D.ietf-dmm-fpc-cpdp].

GL-switch:6 It was already mentioned before that, if there are in-flight packets toward the old anchor while the MN is moving to the new anchor, it may be necessary to buffer these packets and then forward to the new anchor after the old anchor knows that the new anchor is ready Here, however, the corresponding FM operations and mobility message parameters as described in Section 3.2.2 (FM-buffer) can be realized by the internal operations in the control plane together with signaling between the control plane and distributed data plane. These signaling may be realized by the appropriate messages in [I-D.ietf-dmm-fpc-cpdp].

5.3. IP Prefix/Address Anchor Switching for a Hierarchical Network

The configuration for a hierarchical network is shown in Figures 1(c) and 1(d) in Section 3.1. With centralized control plane, CPA and CPN, with the associated LM and FM-CP are all co-located. There are multiple DPAs (each with FM-DP) in distributed mobility anchoring. In the data plane, there are multiple DPNs (each with FM-DP) hierarchically below each DPA. The DPA at each AR supports forwarding to the DPN at each of a number of forwarding switches (FW’s). A mobility event in this configuration belonging to distributed mobility management will be deferred to Section 5.4.

In this distributed mobility configuration, a mobility event involving change of FW only but not of AR as shown in Figure 11 may still belong to centralized mobility management and may be supported using PMIPv6. This configuration of network-based mobility is also applicable to host-based mobility with the modification for the MN directly taking the role of DPN and CPN, and the corresponding centralized mobility event may be supported using MIPv6.

In Figure 11, the IP prefix allocated to the MN is anchored at the access router (AR) supporting indirection to the old FW to which the MN was originally attached as well as to the new FW to which the MN has moved.

The realization of LM may be the binding between the IP prefix/address of the flow used by the MN and the IP address of the DPN to which MN has moved. The implementation of FM to enable change of FW without changing AR may be accomplished using tunneling between the AR and the FW as described in [I-D.korhonen-dmm-local-prefix] and in [I-D.templin-aerolink] or using some other L2 mobility mechanism.

```
Net1                                                   Net2
+----------------------------------------------------------------------+
|                            CPA,CPN:                                  |
|                            LM:IP1<-->IPn2                            |
|                            FM-CP                                     |
+----------------------------------------------------------------------+


                           +---------------+
                           |AR1            |
                           +---------------+
                           |DPA(IPa1):     |
                           |anchors IP1    |
                           |FM:DHCPv6-PD   |
                           +---------------+


+---------------+                                      +---------------+
|FW1            |                                      |FW2            |
+---------------+                 move                 +---------------+
|DPN(IPn1):     |               =======>               |DPN(IPn2):     |
+---------------+                                      +---------------+


+...............+                                      +---------------+
.MN(IP1)        .                 move                 |MN(IP2)        |
.flow(IP1,...)  .               =======>               |flow(IP1,...)  |
+...............+                                      +---------------+
```

Figure 11. Mobility without involving change of IP anchoring in a network in which the IP prefix allocated to the MN is anchored at an AR which is hierarchically above multiple FWs to which the MN may connect.

5.3.1. Additional Guidelines for IPv6 Nodes: No Anchoring Change with a Hierarchical Network

The configuration guideline ( ) for a hierarchical network or network slice with centralized control plane and supporting a mix of flows requiring and not requiring IP mobility support is:

GL-cfg:5 Multiple instances of DPAs (at access routers) which are providing IP prefix to the MNs are needed to provide distributed mobility anchoring according to Figure 2(a) or Figure 2(b)in Section 3.1.2 with centralized control plane for a hierarchical network.

The appropriate IPv6 nodes (CPA, DPA) are to be implemented the mobility functions LM and FM as described respectively in LM-cfg:3 or LM-cfg:4 and FM-cfg:2 in Section 3.2.

Even when the mobility event does not involve change of anchor, it is still necessary to distinguish whether a flow needs IP mobility support.

The GL-mix guidelines in Section 4.1.1 and in Section 4.2.1 for the IPv6 nodes for a network or network slice supporting a mix of flows requiring and not requiring IP mobility support apply here. The guidelines (GL-switch) in Section 5.1.1 and in Section 5.2.1 also apply here. In addition, the following are required.

GL-switch:7 Here, the LM operations and mobility message parameters described in Section 3.2.1 provides information of which IP prefix from its FW needs to be used by a flow using which new FW. The anchor operations to properly forward the packets of a flow described in the FM operations and mobility message parameters (FM-path, FM-path-ind, FM-cpdp in Section 3.2.2) may be realized with PMIPv6 protocol ([I-D.korhonen-dmm-local-prefix]) or with AERO protocol ([I-D.templin-aerolink]) to tunnel between the AR and the FW.

5.4. IP Prefix/Address Anchor Switching for a Hierarchical Network

The configuration for the hierarchical network is again shown in Figures 1(c) and 1(d) in Section 3.1. Again, with centralized control plane, CPA and CPN, with the associated LM and FM-CP are all co-located. There are multiple DPAs (each with FM-DP) in distributed mobility anchoring. In the data plane, there are multiple DPNs (each with FM-DP) hierarchically below each DPA. The DPA at each AR supports forwarding to the DPN at each of a number of forwarding switches (FW’s).

A distributed mobility event in this configuration involves change from a previous DPN which is hierarchically under the previous DPA to a new DPN which is hierarchically under a new DPA. Such an event involving change of both DPA and DPN is shown in Figure 12.

```
Net1                                                   Net2
+----------------------------------------------------------------------+
|                            CPA,CPN:                                  |
|                            LM:IP1<-->IPa2,IPn2                       |
|                            FM-CP                                     |
+----------------------------------------------------------------------+

                            +---------------+
                            |Aggregate Point|
                            |---------------|
                            |FM,    LM      |
                            +---------------+


+---------------+                                      +---------------+
|AR1            |                                      |AR2            |
+---------------+                                      +---------------+
|DPA(IPa1):     |                                      |DPA(IPa2):     |
|anchors IP1    |                 move                 |anchors IP2,IP1|
|FM:DHCPv6-PD   |               =======>               |FM:DHCPv6-PD   |
+---------------+                                      +---------------+


+---------------+                                      +---------------+
|FW1            |                                      |FW2            |
+---------------+                 move                 +---------------+
|DPN(IPn1):     |               =======>               |DPN(IPn2):     |
+---------------+                                      +---------------+


+...............+                                      +---------------+
.MN(IP1)        .                 move                 |MN(IP2,IP1)    |
.flow(IP1,...)  .               =======>               |flow(IP1,...)  |
+...............+                                      +---------------+
```

Figure 12. Mobility involving change of IP anchoring in a network with hierarchy in which the IP prefix allocated to the MN is anchored at an Edge Router supporting multiple access routers to which the MN may connect.

This deployment case involves both a change of anchor from AR1 to AR2 and a network hierarchy AR-FW. It can be realized by a combination of changing the IP prefix/address anchoring from AR1 to AR2 with the mechanism as described in Section 5.2 and then forwarding the packets with network hierarchy AR-FW as described in Section 5.3.

To change AR, AR1 acting as a DHCPv6-PD client may exchange message with the DHCPv6 server to release the prefix IP1. Meanwhile, AR2 acting as a DHCPv6-PD client may exchange message with the DHCPv6 server to delegate the prefix IP1 to AR2.

5.4.1. Additional Guidelines for IPv6 Nodes: Switching Anchor with Hierarchical Network

The configuration guideline (GL-cfg) for a hierarchical network or network slice with centralized control plane described in Section 5.3.1 apply here.

The GL-mix guidelines in Section 4.1.1 and in Section 4.2.1 for the IPv6 nodes for a network or network slice supporting a mix of flows requiring and not requiring IP mobility support apply here.

The guidelines (GL-switch) in Section 5.1.1 and in Section 5.2.1 also apply here to change the anchoring of the IP prefix/address with a centralized control plane.

In addition, the guideline for indirection between the new DPA and the new DPN as described in Section 5.3.1 apply here.

6. Security Considerations

TBD

7. IANA Considerations

This document presents no IANA considerations.

8. Contributors

This document has benefited from other work on mobility solutions using BGP update, on mobility support in SDN network, on providing mobility support only when needed, and on mobility support in enterprise network. These work have been referenced. While some of these authors have taken the work to jointly write this document, others have contributed at least indirectly by writing these drafts. The latter include Philippe Bertin, Dapeng Liu, Satoru Matushima, Peter McCann, Pierrick Seite, Jouni Korhonen, and Sri Gundavelli.

Valuable comments have also been received from John Kaippallimalil, ChunShan Xiong, and Dapeng Liu.

9. References

9.1. Normative References

[I-D.ietf-dmm-deployment-models]
Gundavelli, S. and S. Jeon, "DMM Deployment Models and Architectural Considerations", draft-ietf-dmm-deployment-models-00 (work in progress), August 2016.

[I-D.ietf-dmm-fpc-cpdp]
Liebsch, M., Matsushima, S., Gundavelli, S., Moses, D., and L. Bertz, "Protocol for Forwarding Policy Configuration (FPC) in DMM", draft-ietf-dmm-fpc-cpdp-03 (work in progress), March 2016.

[I-D.ietf-dmm-ondemand-mobility]
Yegin, A., Moses, D., Kweon, K., Lee, J., and J. Park, "On Demand Mobility Management", draft-ietf-dmm-ondemand-mobility-07 (work in progress), July 2016.

[I-D.jhlee-dmm-dnpp]
Lee, J. and Z. Yan, "Deprecated Network Prefix Provision", draft-jhlee-dmm-dnpp-01 (work in progress), April 2016.

[I-D.korhonen-dmm-local-prefix]
Korhonen, J., Savolainen, T., and S. Gundavelli, "Local Prefix Lifetime Management for Proxy Mobile IPv6", draft-korhonen-dmm-local-prefix-01 (work in progress), July 2013.

[I-D.liu-dmm-deployment-scenario]
Liu, V., Liu, D., Chan, A., Lingli, D., and X. Wei, "Distributed mobility management deployment scenario and architecture", draft-liu-dmm-deployment-scenario-05 (work in progress), October 2015.

[I-D.matsushima-stateless-uplane-vepc]
Matsushima, S. and R. Wakikawa, "Stateless user-plane architecture for virtualized EPC (vEPC)", draft-matsushima-stateless-uplane-vepc-06 (work in progress), March 2016.

[I-D.mccann-dmm-flatarch]
McCann, P., "Authentication and Mobility Management in a Flat Architecture", draft-mccann-dmm-flatarch-00 (work in progress), March 2012.

[I-D.mccann-dmm-prefixcost]
McCann, P. and J. Kaippallimalil, "Communicating Prefix Cost to Mobile Nodes", draft-mccann-dmm-prefixcost-03 (work in progress), April 2016.

[I-D.seite-dmm-dma]
Seite, P., Bertin, P., and J. Lee, "Distributed Mobility Anchoring", draft-seite-dmm-dma-07 (work in progress), February 2014.

[I-D.sijeon-dmm-deployment-models]
Jeon, S. and Y. Kim, "Deployment Models for Distributed Mobility Management", draft-sijeon-dmm-deployment-models-03 (work in progress), July 2016.

[I-D.templin-aerolink]
Templin, F., "Asymmetric Extended Route Optimization (AERO)", draft-templin-aerolink-71 (work in progress), September 2016.

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

[RFC3753] Manner, J., Ed. and M. Kojo, Ed., "Mobility Related Terminology", RFC 3753, DOI 10.17487/RFC3753, June 2004, <http://www.rfc-editor.org/info/rfc3753>.

[RFC5213] Gundavelli, S., Ed., Leung, K., Devarapalli, V., Chowdhury, K., and B. Patil, "Proxy Mobile IPv6", RFC 5213, DOI 10.17487/RFC5213, August 2008, <http://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, <http://www.rfc-editor.org/info/rfc6275>.

[RFC7077] Krishnan, S., Gundavelli, S., Liebsch, M., Yokota, H., and J. Korhonen, "Update Notifications for Proxy Mobile IPv6", RFC 7077, DOI 10.17487/RFC7077, November 2013, <http://www.rfc-editor.org/info/rfc7077>.

[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, <http://www.rfc-editor.org/info/rfc7333>.

[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, <http://www.rfc-editor.org/info/rfc7429>.

9.2. Informative References

[Paper-Distributed.Mobility]
Lee, J., Bonnin, J., Seite, P., and H. Chan, "Distributed IP Mobility Management from the Perspective of the IETF: Motivations, Requirements, Approaches, Comparison, and Challenges", IEEE Wireless Communications, October 2013.

[Paper-Distributed.Mobility.PMIP]
Chan, H., "Proxy Mobile IP with Distributed Mobility Anchors", Proceedings of GlobeCom Workshop on Seamless Wireless Mobility, December 2010.

[Paper-Distributed.Mobility.Review]
Chan, H., Yokota, H., Xie, J., Seite, P., and D. Liu, "Distributed and Dynamic Mobility Management in Mobile Internet: Current Approaches and Issues", February 2011.

Authors’ Addresses

H Anthony Chan (editor)
Huawei Technologies
5340 Legacy Dr. Building 3
Plano, TX 75024
USA

Email: h.a.chan@ieee.org

Xinpeng Wei
Huawei Technologies
Xin-Xi Rd. No. 3, Haidian District
Beijing, 100095
P. R. China

Email: weixinpeng@huawei.com

Jong-Hyouk Lee
Sangmyung University
31, Sangmyeongdae-gil, Dongnam-gu
Cheonan 31066
Republic of Korea

Email: jonghyouk@smu.ac.kr

Seil Jeon
Sungkyunkwan University
2066 Seobu-ro, Jangan-gu
Suwon, Gyeonggi-do
Republic of Korea

Email: seiljeon@skku.edu

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

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

Fred L. Templin
Boeing Research and Technology
P.O. Box 3707
Seattle, WA 98124
USA

Email: fltemplin@acm.org

DMM Working Group
Internet-Draft
Intended status: Standards Track
Expires: November 7, 2015

M. Liebsch
NEC
S. Matsushima
Softbank Telecom
S. Gundavelli
Cisco
D. Moses
Intel Corporation

May 6, 2015

Protocol for Forwarding Policy Configuration (FPC) in DMM
draft-ietf-dmm-fpc-cpdp-00.txt

Abstract

The specification as per this document supports the separation of the Control-Plane for mobility- and session management from the actual Data-Plane. The protocol semantics abstract from the actual details for the configuration of Data-Plane nodes and apply between a Client function, which is used by an application of the mobility Control-Plane, and an Agent function, which is associated with the configuration of Data-Plane nodes according to the policies issued by the mobility Control-Plane. The scope of the policies comprises forwarding rules and treatment of packets in terms of encapsulation, IP address re-writing and QoS. Additional protocol semantics are described to support the maintenance of the Data-Plane path.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

One objective of the Distributed Mobility Management (DMM) WG is the separation of the mobility management Control- and Data-Plane to enable flexible deployment, such as decentralized provisioning of Data-Plane nodes (DPN). Data-Plane nodes can be configured to function as anchor for a registered Mobile Node's (MN) traffic, others can be configured to function as Mobile Access Gateway (MAG) as per the Proxy Mobile IPv6 protocol [RFC5213] or a Foreign Agent

(FA) as per the Mobile IPv4 protocol [RFC3344]. Requirements for DMM have been described in [RFC7333], whereas best current practices for DMM are documented in [RFC7429].

The Data-Plane must provide a set of functions to the Mobility Control-Plane, such as support for encapsulation, IP address re-writing, QoS differentiation and traffic shaping. In addition, the configuration of forwarding rules must be provided. These requirements are met by various transport network components, such as IP switches and routers, though configuration semantics differs between them.

Forwarding Policy Configuration (FPC) as per this document enables the configuration of any Data-Plane node and type by the abstraction of configuration details and the use of common configuration semantics. The protocol using the FPC semantics is deployed between a Client function, which is associated with the Mobility Management Control-Plane, and an Agent function. The Agent function enforces the Data-Plane configuration and can be present on a transport network controller or co-located with a Data-Plane node. The Agent applies the generalized configuration semantics to configuration, which is specific to the Data-Plane node and type. The Mobility Control-Plane can select one or multiple DPNs which suit the MN’s mobility management without the need to handle each node’s routing- or switching tables and local interface configurations for potentially many routers serving the Data-Plane, but enforce the policies for traffic treatment and forwarding through the FPC Client and the FPC Agent functions.

2. Conventions and 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 [RFC2119].

3. Model for Policy-based DMM Network Control

3.1. Reference Architecture for DMM Forwarding Policy Configuration

The DMM Forwarding Policy Configuration (FPC) protocol enables DMM use cases in deployments with separated Control-/Data-Plane and is used by applications of the Mobility Control-Plane to enforce rules for forwarding and traffic treatment in the Data-Plane. Figure 1 depicts an exemplary use case where downlink traffic from a Correspondent Node (CN) towards a Mobile Node (MN) traverses multiple DPNs, each applying policies as per the Control-Plane’s request. Policies in the one or multiple DPNs can result in traffic steering according to a host-route, packet scheduling and marking according to a subscriber’s QoS profile, or forwarding rules (e.g. encapsulation within GRE or GTP-U tunnel).

```
                           +--------------------------+
                           |     Mobility Control     |
                           +--------------------------+
                            |             |         |
                            |             |         |
                            |             |         |
                     \ /    V             V         V
            +--+     -o-  +---+         +---+     +---+   +--+
            |MN| ---- |---|DPN|<========|DPN|<----|DPN|<--|CN|
            +--+      |   +---+         +---+     +---+   +--+
                          Rules:       Rules:     Rules:
                          Decap,       Encap,     host-route
                          Forward      Forward,
                                       qos
```

Figure 1: Exemplary illustration of a use case for DMM traffic steering and policy enforcement at Data Plane Nodes (DPN)

Mobility Control-Plane functions have the following roles in common:

- Tracking an MN’s location

- Accept requests to set up and maintain mobility-related Data-Plane path between DPNs, taking QoS attributes into account. Such requests can be issued through mobility protocols, such as Proxy Mobile IPv6, and the associated operation with remote Mobility Control-Plane functions.

- Become aware of different DPNs that provide the required Data-plane functions to the Mobility Control-Plane and can be used for mobility traffic forwarding and treatment

- Monitor the DPNs’ operation and handle exceptions, e.g. the detection of a partial DPN failure and the diversion of traffic through a different DPN

- Maintain consistency between multiple DPNs which enforce policy rules for an MN

Mobility Data-Plane functions have the following roles in common:

- Forward and treat traffic according to the policies and directives sent by the Mobility Control-Plane

- Provide status (e.g. load, health, statistics and traffic volume) information on request

- Participate in the process for topology acquisition, e.g. by exposing relevant topological and capability information, such as support for QoS differentiation and supported encapsulation protocols

The protocol for DMM FPC applies to the interface between an FPC Client function and an FPC Agent function, as depicted in Figure 2. The FPC Client function is associated with an application function of the mobility management Control-Plane, e.g. a Local Mobility Anchor Control-Plane function as per the Proxy Mobile IPv6 protocol. The FPC Agent function processes the FPC protocol semantics and translates them into configuration commands as per the DPN’s technology. In one example, an FPC Agent can be co-located with a Transport Network Controller, which enforces forwarding rules on a set of SDN switches. In another example, the Agent can be co-located with a single router to directly interact with interface management and the router’s RIB Manager. The mapping of the common FPC semantics and policy description as per this specification to the configuration commands of a particular DPN is specific to the DPN’s technology and the Agent’s implementation.

```
+-------------------------+
| Mobility Control-Plane  |
|                         |
|+--------[API]----------+|
||  FPC Client Function  ||
|+----------^------------+|
+-----------|-------------+
            |
            | DMM FPC protocol
            |
+-----------|-------------+
|+----------v------------+|
||  FPC Agent Function   ||
|+-----------------------+|
|                         |
|  DPN Configuration API  |
+-------------------------+
```

Figure 2: Illustration of the functional reference architecture for DMM Forwarding Policy Configuration (FPC)

3.2. Generalized Rules on the Client-Agent-Interface

To abstract configuration details of an IP switch or IP router on the FPC protocol interface, this specification adopts the model of logical gates (Ports) to bind certain properties, such as a QoS policy. Additional properties can be bound to the same logical Port, e.g. encapsulation of packets, being directed to that logical Port, in a GRE tunnel. The remote tunnel endpoint is configured as part of the property bound to that logical Port. All traffic, which has a forwarding rule in common and should be forwarded according to the properties bound to a particular Port, can be referred to that Port by configuration of a forwarding rule. Multiple IP flows or even aggregated traffic being destined to a given IP prefix can be directed to that logical Port and experiences the same treatment according to the configured properties and forwarding characteristics. Aggregated or per-Host/per-Flow traffic can be identified by a longest prefix match or a Traffic Selector respectively.

Figure 3 illustrates the generic policy configuration model as used between an FPC Client function and an FPC Agent function.

```
    <prefix 1>     <IP flow 1>  <host src IP 1>    +-------------------+
  <host dst IP 1>  <IP flow 2>   <prefix 2>        |     Bind 1..M     |
         |         <IP flow 3>       |             | traffic templates |
         |             |             |             |  to each logical  |
         |             |             |             |       port        |
         |             |             |             +-------------------+
         v             v             v
      <PORT_1>------<PORT_2>------<PORT_3>--- ...  [logical ports space]
         |             |             |
         +--PROP_1.1   +--PROP_2.1   +--PROP_3.1   +-------------------+
         |                           |             |     Bind 1..N     |
         +--PROP_1.2                 +--PROP_3.2   |     properties    |
         |                                         |  to each logical  |
         +--PROP_1.3                               |       port        |
                                                   +-------------------+
```

Figure 3: Illustration of generalized rules

3.3. Role of the DMM FPC Client Function

The DMM FPC Client function includes the following tasks:

- Per mobility management transaction or relevant event, build one or multiple Control messages/attributes to control policies on one or multiple DPA(s) according to the application’s directives

- Treat a DPN’s policy rules (encapsulation, address re-write, QoS, traffic monitoring) on the basis of properties being bound to logical ports (similar to the bearer concept in cellular networks)

- Build, modify or delete logical ports as needed

- Bind associated policy rules as one or multiple properties to a logical port

- Treat forwarding rules (e.g. per-IP flow, per-MN, per-IP, per-prefix) on the basis of logical ports

- Send each generated message to the DMM FPC Agent associated with the identified DPN

- Keep record of the policy rules/port information and the associated DPN and FPC Agent Function

- Process received Response, Notification and Query messages issued by a DMM FPC Agent Function and notify the application

3.4. Role of the DMM FPC Agent Function

The DMM FPC Agent function includes the following tasks:

- Process the received Control messages issued by a DMM FPC Client Function

- Unambiguously match each logical port with an associated physical port or interface at the identified DPN

- Apply the received properties to local configuration (e.g. encapsulation, NA(P)T, traffic prioritization and scheduling) on the identified DPN according to the DPN’s technology

- Monitor scheduled events (e.g. failure or missing rule) and issue an associated message to the FPC Client Function (NOTIFICATION, QUERY)

4. Protocol Messages and Semantics

4.1. Protocol Messages

Message	Description
Messages issued by the FPC Client	
PRT_ADD	Add a logical port
PRT_DEL	Delete an existing logical port
PROP_ADD	Add a property to a logical port
PROP_MOD	Modify a property of a logical port
PROP_DEL	Remove and delete a property from a logical port
RULE_ADD	Add forwarding rule by binding traffic descriptor to a logical port
RULE_MOD	Modify existing forwarding rule by changing the traffic descriptor bound to a logical port
RULE_DEL	Delete a forwarding rule
EVENT_REG	Register an event at an Agent, which is to be monitored by the Agent and to be reported
PROBE	Probe the status of a registered event
Messages issued by the FPC Agent	
NOTIFY	Notify the Client about the status of a monitored attribute at any event kind (periodic / event trigger / probed)
QUERY	Query the Client about missing rules/states

Figure 4: Protocol Messages

4.2. Protocol Attributes

Protocol messages as per Section 4.1 carry attributes to identify an FPC Client- or Agent function, as well as a DPN, logical ports and configuration data. Furthermore, attributes are carried to manage logical ports and describe properties associated with a logical port, as well as to describe per-host-, aggregate or IP flow traffic and refer to a logical port as forwarding information.

This document specifies attributes from the following categories:

- Identifier attributes

- Properties

- Property-specific attributes

- Traffic descriptors

Note on the list of attributes: The list of attributes is not yet complete.

Note on Format Clarification: Meant to provide a first idea on the format and number space and indicates length (bit) and semantics of key information fields.

Attribute	Format Clarification	Description
Identifiers		
PRT_ID	[16,PTR_ID]	Identifies a logical Port
PRT_PROP_ID	[16,PRT_ID] [8,PROP_ID]	Identifies a logical Port and one of its properties
CLI_ID	[8, Carrier ID] [8, Network ID] [16, Client ID]	Identifies an FPC Client function
AGT_ID	[8, Carrier ID] [8, Network ID] [16, Agent ID]	Identifies an FPC Agent function
DPN_ID	[8, Carrier ID] [8, Network ID] [16, DPN ID]	Identifies a Data Plane Node (DPN)
EVENT_ID	[16, Event ID]	Identifies a registered event

Figure 5: Protocol Attributes: Identifiers

Attribute	Format Clarification	Description
Properties		
PROP_TUN	[type][src][dst]	Property Encapsulation, indicates type GRE, IP, GTP
PROP_REWR	TBD	Property NAT
PROP_QOS	TBD	Property QoS
PROP_GW	[ip address next hop]	Property Next Hop

Figure 6: Protocol Attributes: Properties

Attribute	Format Clarification	Description
Property-specific		
IPIP_CONF		IP-encapsulation configuration attribute
GRE_CONF	[prototype][seq-#] [key]..	GRE_encapsulation configuration attribute
GTP_CONF	[TEID_local] [TEID_remote] [seq-#]..	GTP-U encapsulation configuration attribute

Figure 7: Protocol Attributes: Property-specific

4.3. Protocol Operation

The following list comprises a more detailed description of each message’s semantic.

- PRT_ADD - Issued by a Client to add a new logical port at an Agent, to which traffic can be directed. An Agent receiving the PRT_ADD message should identify the new logical port according to the included port identifier (PRT_ID). In case the DPN holds already a registration for a logical port with the same identifier, the Agent should throw an error message to the Client. Otherwise the Agent should add a new logical port into its conceptual data structures using the port identifier as key.

- PRT_DEL - Used by a Client to delete an existing logical port. An Agent receiving such message should delete all properties associated with the identified port.

- PROP_ADD - Used by the Client to add a new property to an existing logical port. The property is unambiguously identified through a property identifier (PRT_PROP_ID). All traffic, which is directed to this logical port, experiences the existing and newly added property.

- PROP_MOD - Used by a Client to modify an existing property. For example, a tunnel property can be changed to direct traffic to a different tunnel endpoint in case of an MN’s handover

- PROP_DEL - Used by a Client to delete one or multiple properties, each being identified by a property identifier.

- RULE_ADD - Used by a Client to add a forwarding rule and direct traffic towards a logical port. The rule add command must unambiguously identify aggregated traffic (longest prefix), per host IP traffic or per-flow traffic in the RULE_ADD command and bind the identified traffic to a logical port. An Agent receiving a RULE_ADD command must add the rule to its local conceptual data structures and apply commands for local configuration to add the new forwarding rule on the DPN. Multiple forwarding rules, each identifying different traffic, can direct traffic to the same logical port. All traffic being directed to this logical port will then experience the same properties.

- RULE_MOD - Used by a Client to modify an existing forwarding rule. An Agent receiving such message should apply commands for local configuration to update the forwarding rule on the DPN.

- RULE_DEL - Used to delete an existing forwarding rule on a DPN. The Agent receiving such message should delete the rules from its local conceptual data structures and apply commands for local configuration to remove the forwarding rule on the DPN.

- EVENT_REG - Used by a Client to register an attribute, which is to be monitored, at an Agent. The EVENT_REG provides an attribute to the Agent as well as a reporting kind. The Agent should register the event and an event identifier in the local conceptual data structures. The Agent should start monitoring the registered attribute (e.g. load) and notify the Client about the status according to the registered reporting kind (periodic, event trigger, probed). In case of a periodic reporting kind, the Agent should report the status of the attribute each configured interval using a NOTIFY message. The reporting interval is provided with the EVENT_REG message. In case of an event triggered reporting kind, the Agent should report the status of the attribute in case of a triggered event, e.g. the monitored attribute’s value exceeds a given threshold. The threshold is provided with the EVENT_REG message. In case of probed reporting, the Agent receives a PROBE message and should report the status of a monitored attributes to the Client by means of a NOTIFY message.

- PROBE - Used by a Client to retrieve information about a previously registered event. The PROBE message should identify one or more events by means of including the associated event identifier. An Agent receiving a PROBE message should send the requested information for each event in a single or multiple NOTIFY messages.

- NOTIFY - Used by an Agent to report the status of an event to a Client.

- QUERY - Used by an Agent to request an update of logical port properties via a Client.

Figure 8 illustrates an exemplary session life-cycle based on Proxy Mobile IPv6 registration via MAG Control-Plane function 1 (MAG-C1) and handover to MAG Control-Plane function 2 (MAG-C2). Edge DPN1 represents the Proxy CoA after attachment, whereas Edge DPN2 serves as Proxy CoA after handover.

```
                                                 +-------Router--------+
                         +-----------+           |+-------+ +---------+|
 +------+ +------+     +-----+ FPC   |            | FPC   | |  Anchor |
 |MAG-C1| |MAG-C2|     |LMA-C| Client|            | Agent | |   DPN   |
 +------+ +------+     +-----+-------+            +-------+ +---------+
 [MN attach]  |            |                          |           |
    |-------------PBU----->|                          |           |
    |         |            |----(1)-PRT_ADD---------->|           |
    |         |            |        [PRT_ID]          |           |
    |         |            |                          |           |
    |         |            |--(2)---PROP_ADD--------->|           |
    |         |            |   [PROP_ID,PROP_TUN]     |--tun1 up->|
    |         |            |                          |           |
    |         |            |--(3)---PROP_ADD--------->|           |
    |         |            |    [PROP_ID,PROP_QOS]    |--tc qos-->|
    |<------------PBA------|--(4)----RULE_ADD-------->|           |
    | +----+  |            |       [HNP,PRT_ID]       |-route add>|
    | |Edge|  |            |                          |           |
    | |DPN1|  |            |                          |           |
    | +----+  |            |                          |           |
    |   |                                                         |
    |   |-=======================================================-|
    |                      |                          |           |
    |   [MN handover]      |                          |           |
    |         |---PBU ---->|                          |           |
    |         |            |--(5)---PROP_MOD--------->|           |
    |         |<--PBA------|   [PROP_ID,PROP_TUN]     |-tun1 mod->|
    |         |            |                          |           |
    |         |  +----+    |                          |           |
    |         |  |Edge|    |                          |           |
    |         |  |DPN2|    |                          |           |
    |         |  +----+    |                          |           |
    |         |    |       |                          |           |
    |         |    |-============================================-|
    |         |            |                          |           |
```

Figure 8: Exemplary Message Sequence (focus on FPC reference point)

After reception of the Proxy Binding Update (PBU) at the LMA Control-Plane function (LMA_C), the LMA-C selects a suitable DPN, which serves as Data-Plane anchor to the MN’s traffic. The LMA-C adds a new logical port to the DPN to treat the MN’s traffic (1) and includes a Port Identifier (PRT_ID) to the PRT_ADD command. The LMA-C identifies the selected Anchor DPN by including the associated DPN identifier.

Subsequently, the LMA-C adds properties to the new logical port. One property is added (2) to specify the forwarding tunnel type and endpoints (Anchor DPN, Edge DPN1). Another property is added (3) to specify the QoS differentiation, which the MN’s traffic should experience. At reception of the properties, the FPC Agent calls local router commands to enforce the tunnel configuration (tun1) as well as the traffic control (tc) for QoS differentiation. After configuration of port properties have been completed, the LMA can configure the enforcement of the MN’s traffic by adding a rule (RULE_ADD) to forward traffic destined to the MN’s HNP to the new logical port (4). At the reception of the forwarding rule, the Agent applies a new route to forward all traffic destined to the MN’s HNP to the configured tunnel interface (tun1).

During handover, the LMA-C receives an updating PBU from the handover target MAG-C2. The PBU refers to a new Data-Plane node (Edge DPN2) to represent the new tunnel endpoint. The LMA-C sends a PROP_MOD message (5) to the Agent to modify the existing tunnel property of the existing logical port and to update the tunnel endpoint from Edge DPN1 to Edge DPN2. At reception of the PROP_MOD message, the Agent applies local configuration commands to modify the tunnel.

To reduce the number of protocol handshakes between the LMA-C and the DPN, the LMA-C can append property (PROP_TUN, PROP_QOS) and rules (prefix info HNP) attributes to the PRT_ADD message, as illustrated in Figure 9

```
                         +-----------+             +-------+ +---------+
 +------+ +------+     +-----+ FPC    |             | FPC   | |  Anchor |
 |MAG-C1| |MAG-C2|     |LMA-C| Client|             | Agent | |   DPN   |
 +------+ +------+     +-----+-------+             +-------+ +---------+
 [MN attach]  |            |                           |           |
    |-------------PBU----->|                           |           |
    |         |            |----(1)-PRT_ADD----------->|           |
    |         |            | [PRT_ID,PROP_ID,PROP_TUN, |--tun1 up->|
    |<------------PBA------|         PROP_ID,PROP_QOS, |--tc qos-->|
    |         |            |         HNP]              |-route add>|
    | [Edge]-=====================================================-|
    | [DPN1|  |            |                           |           |
    |         |            |                           |           |
    |   [MN handover]      |                           |           |
    |         |---PBU ---->|                           |           |
    |         |            |---------PROP_MOD--------->|           |
    |         |<--PBA------|    [PROP_ID,PROP_TUN]     |-tun1 mod->|
    |         |            |                           |           |
    |         | [Edge]-===========================================-|
    |         | [DPN2]     |                           |           |
```

Figure 9: Example: Sequence for Message Aggregation (focus on FPC reference point)

5. Conceptual Data Structures

An FPC Client must keep record about the logical ports, each port’s properties as well as configured rules as per the Mobility Control-Plane function’s request. Such information must be maintained for each Agent, with which the Client communicates. In case the Mobility Control-Plane function identifies a particular DPN at which the policies should be enforced, the Client must associate the DPN identifier with the logical port configuration.

According to the FPC Agent’s role, the Agent translates the generalized model for policy configuration and forwarding rules into semantics and commands for local configuration, which is specific to a DPN. Keeping a local record of DPN configuration attributes/values is implementation specific and out of scope of this document.

Description of detailed data structures and information to be recorded and maintained by an FPC Client and an FPC Agent are TBD and will be added to a revision of this initial document.

6. Security Considerations

Detailed protocol implementations for DMM Forwarding Policy Configuration must ensure integrity of the information exchanged between an FPC Client and an FPC Agent. Required Security Associations may be derived from co-located functions, which utilize the FPC Client and FPC Agent respectively.

7. IANA Considerations

This document provides an information model for DMM Forwarding Policy Configuration. Detailed protocol specifications for DMM Forwarding Policy Configuration will follow the information model as per this document and can be based on, for example, ReST-like or binary protocol formats. Such protocol-specific details will be described in separate documents and may require IANA actions.

8. Work Team Participants

Participants in the FPSM work team discussion include Satoru Matsushima, Danny Moses, Sri Gundavelli, Marco Liebsch, Pierrick Seite, Alper Yegin, Carlos Bernardos, Charles Perkins and Fred Templin.

9. References

9.1. Normative References

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

[RFC7333] Chan, H., Liu, D., Seite, P., Yokota, H., and J. Korhonen, "Requirements for Distributed Mobility Management", RFC 7333, August 2014.

[RFC7429] Liu, D., Zuniga, JC., Seite, P., Chan, H., and CJ. Bernardos, "Distributed Mobility Management: Current Practices and Gap Analysis", RFC 7429, January 2015.

9.2. Informative References

[RFC3344] Perkins, C., "IP Mobility Support for IPv4", RFC 3344, August 2002.

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

Appendix A. YANG Data Model for the FPC Protocol

This appendix provides (so far experimental) formating of some FPC protocol components adopting YANG data modeling. The current FPC information model as per this initial draft version will experience extensions, as it is not yet complete, and may experience changes that need to be reflected in the data model. Whether a detailed data model will be included in this document or solely an information model will be adopted by this document and a detailed data model will be part of a separate document is currently being discussed.

```
   module ietf-dmm-fpcp {
       namespace "urn:ietf:params:xml:ns:yang:dmm-fpcp";
       prefix fpcp;

       import ietf-inet-types { prefix inet; }

       description
       "This module contains YANG definition for
        Forwarding Policy Configuration Protocol.(FPCP)";

       revision 2015-03-09 {}

       typedef fpcp-port-id {
           description "PRT_ID";
           type uint16;
       }

       typedef fpcp-property-id {
           description "PROP_ID";
           type uint8;
       }

       identity tunnel-type {
           description
           "Base identity from which specific use of
            tunnels are derived.";
       }

       identity fpcp-tunnel-type {
           base "tunnel-type";
           description
           "Base identity from which specific tunnel
            types in FPCP uses are derived.";
       }

       identity ip-in-ip {
           base "fpcp-tunnel-type";
           description "IP-in-IP tunnel";
       }

       identity gtp {
           base "fpcp-tunnel-type";
           description "GTP-U tunnel";
       }

       identity gre {
           base "fpcp-tunnel-type";
           description "GRE tunnel";
       }

       identity ip-protocol {
           description
           "Base identity from which specific
            IP protocol types are derived.";
       }

       identity qos-type {
           description
           "Base identity from which specific
            uses of QoS types are derived.";
       }

       identity fpcp-qos-type {
           base "qos-type";
           description
           "Base identity from which specific
            QoS types in FPCP uses are derived.";
       }

       identity fpcp-qos-type-high {
           base "fpcp-qos-type";
           description
           "An example FPCP QoS Type for high quality class.
            FPCP supported QoS classes are TBD.";
       }

       identity fpcp-qos-type-middle {
           base "fpcp-qos-type";
           description
           "An example FPCP QoS Type for middle quality class.
            FPCP supported QoS classes are TBD.";
       }

       identity fpcp-qos-type-low {
           base "fpcp-qos-type";
           description
           "An example FPCP QoS Type for low quality class.
            FPCP supported QoS classes are TBD.";
       }

       grouping fpcp-client {
           description "CLI_ID to identify FPCP Client";
           leaf carrier-id {
               type uint8;
           }
           leaf network-id {
               type uint8;
           }
           leaf client-id {
               type uint16;
               mandatory true;
           }
       }

       grouping fpcp-agent {
           description "AGT_ID to identify FPCP Agent";
           leaf carrier-id {
               type uint8;
           }
           leaf network-id {
               type uint8;
           }
           leaf agent-id {
               type uint16;
               mandatory true;
           }
       }

       grouping dpn {
           description "DPN_ID to identify Data-Plane Node";
           leaf carrier-id {
               type uint8;
           }
           leaf network-id {
               type uint8;
           }
           leaf dpn-id {
               type uint16;
               mandatory true;
           }
       }
       grouping port-property-id {
           description "PRT_PROP_ID";
           leaf port-id {
               mandatory true;
               type fpcp-port-id;
           }
           leaf property-id {
               type fpcp-property-id;
               mandatory true;
           }
       }

       grouping tunnel-endpoints {
           description
           "PROP_TUN property as a set of tunnel endpoints";
           leaf tunnel-type {
               type identityref {
                   base "fpcp-tunnel-type";
               }
           }
           leaf remote-address {
               type inet:ip-address;
           }
           leaf local-address {
               type inet:ip-address;
           }
       }

       grouping gtp-attributes {
           description
           "GTP_CONF as GTP tunnel specific attributes";
           leaf remote-teid {
               type uint32;
           }
           leaf local-teid {
               type uint32;
           }
       }

       grouping gre-attributes {
           description
           "GRE_CONF as GRE tunnel specific attribute";
           leaf key {
               type uint32;
           }
       }

       grouping fpcp-identifier-attributes {
           description
           "Identifiers of protocol attributes";
           leaf port-id {
               type fpcp-port-id;
           }
           container client {
               uses fpcp-client;
           }
           container agent {
               uses fpcp-agent;
           }
           list nodes {
               key dpn-id;
               uses dpn;
           }
       }

       grouping fpcp-traffic-descriptor {
           description
           "Traffic descriptor group collects parameters to
            identify target traffic flow and apply QoS policy";
           leaf destination-ip {
               type inet:ip-prefix;
           }
           leaf source-ip {
               type inet:ip-prefix;
           }
           leaf protocol {
               type identityref {
                   base "ip-protocol";
               }
           }
           leaf destination-port {
               type inet:port-number;
           }
           leaf source-port {
               type inet:port-number;
           }
           leaf qos {
               type identityref {
                   base "fpcp-qos-type";
               }
           }
       }

       grouping fpcp-port-properties {
           description
           "A set of port property attributes";
           leaf property-id {
               type fpcp-property-id;
           }
           list next-hops {
               container endpoints {
                   uses tunnel-endpoints;
               }
               choice tunnel {
                   case gtp-u {
                       when "tunnel-type = ’gtp’";
                       uses gtp-attributes;
                   }
                   case gre {
                       when "tunnel-type = ’gre’";
                       uses gre-attributes;
                   }
               }
           }
       }


       // Port Entries

       container port-entries {
           description
           "This container binds set of traffic-descriptor and
            port properties to a port and lists them as a port entry.";
           list port-entry {
               key port-id;
               container identifier {
                   uses fpcp-identifier-attributes;
               }
               container trafic-descriptor {
                   uses fpcp-traffic-descriptor;
               }
               list properties {
                   uses fpcp-port-properties;
               }
           }
       }

       // PRT_ADD

       rpc port_add {
           description "PRT_ADD";
           output {
               list fpcp-port-entry {
                   uses fpcp-identifier-attributes;
```

```
              }
          }
      }

      // PRT_DEL

      rpc port_delete {
          description "PRT_DEL";
          input {
              leaf deleting-port {
                  type fpcp-port-id;
              }
          }
      }

      // PROP_ADD

      rpc port_property_add {
          description "PROP_ADD";
          input {
              leaf target-port {
                  type fpcp-port-id;
                  mandatory true;
              }
              container port-properties {
                  uses fpcp-port-properties;
              }
          }
      }

      // PROP_MOD

      rpc port_property_modify {
          description "PROP_MOD";
          input {
              leaf target-port {
                  type fpcp-port-id;
                  mandatory true;
              }
              container port-properties {
                  uses fpcp-port-properties;
              }
          }
      }

      // PROP_DEL

      rpc port_property_delete {
          description "PROP_DEL";
          input {
              container deleting-property {
                  uses port-property-id;
              }
          }
      }

      // RULE_ADD

      rpc rule_add {
          description
          "TBD for input parameters of which RULE_ADD includes
           but now just traffic-descriptor.";
          input {
              leaf target-port {
                  type fpcp-port-id;
                  mandatory true;
              }
              container port-properties {
                  uses fpcp-traffic-descriptor;
              }
          }
      }

      // RULE_MOD

      rpc rule_modify {
          description
          "TBD for input parameters of which RULE_MOD includes
           but now just traffic-descriptor.";
          input {
              leaf target-port {
                  type fpcp-port-id;
                  mandatory true;
              }
              container port-properties {
                  uses fpcp-traffic-descriptor;
              }
          }
      }

      // RULE_DEL

      rpc rule_delete {
          description
          "TBD for input parameters of which RULE_DEL includes
           but now just traffic-descriptor.";
          input {
              leaf target-port {
                  type fpcp-port-id;
                  mandatory true;
              }
              container port-properties {
                  uses fpcp-traffic-descriptor;
              }
          }
      }

      // EVENT_REG

      rpc event_register {
          description
          "TBD for registered parameters included in EVENT_REG.";
      }

      // PROBE

      rpc probe {
          description
          "TBD for retrieved parameters included in PROBE.";
      }

      // NOTIFY

      notification notify {
          description
          "TBD for which status and event are reported to client.";
      }
  }
```

Figure 10: FPC YANG Data Model

Authors' Addresses

Marco Liebsch
NEC Laboratories Europe
NEC Europe Ltd.
Kurfuersten-Anlage 36
D-69115 Heidelberg
Germany

Phone: +49 6221 4342146
Email: liebsch@neclab.eu

Satoru Matsushima
Softbank Telecom
1-9-1,Higashi-Shimbashi,Minato-Ku
Tokyo 105-7322
Japan

Email: satoru.matsushima@g.softbank.co.jp

Sri Gundavelli
Cisco
170 West Tasman Drive
San Jose, CA 95134
USA

Email: sgundave@cisco.com

Danny Moses

Email: danny.moses@intel.com

DMM Working Group
Internet-Draft
Intended status: Standards Track
Expires: 27 March 2021

S. Matsushima
SoftBank
L. Bertz
Sprint
M. Liebsch
NEC
S. Gundavelli
Cisco
D. Moses
Intel Corporation
C.E. Perkins
Futurewei

23 September 2020

Protocol for Forwarding Policy Configuration (FPC) in DMM
draft-ietf-dmm-fpc-cpdp-14

Abstract

This document describes a way, called Forwarding Policy Configuration (FPC) to manage the separation of data-plane and control-plane. FPC defines a flexible mobility management system using FPC agent and FPC client functions. A FPC agent provides an abstract interface to the data-plane. The FPC client configures data-plane nodes by using the functions and abstractions provided by the FPC agent for the data-plane nodes. The data-plane abstractions presented in this document are extensible in order to support many different types of mobility management systems and data-plane functions.

Status of This Memo

This Internet-Draft will expire on 27 March 2021.

Copyright Notice

Table of Contents

1. Introduction

This document describes Forwarding Policy Configuration (FPC), a system for managing the separation of control-plane and data-plane. FPC enables flexible mobility management using FPC client and FPC agent functions. A FPC agent exports an abstract interface representing the data-plane. To configure data-plane nodes and functions, the FPC client uses the interface to the data-plane offered by the FPC agent.

Control planes of mobility management systems, or related applications which require data-plane control, can utilize the FPC client at various levels of abstraction. FPC operations are capable of directly configuring a single Data-Plane Node (DPN), as well as multiple DPNs, as determined by the data-plane models exported by the FPC agent.

A FPC agent represents the data-plane operation according to several basic information models. A FPC agent also provides access to Monitors, which produce reports when triggered by events or FPC Client requests regarding Mobility Contexts, DPNs or the Agent.

To manage mobility sessions, the FPC client assembles applicable sets of forwarding policies from the data model, and configures them on the appropriate FPC Agent. The Agent then renders those policies into specific configurations for each DPN at which mobile nodes are attached. The specific protocols and configurations to configure a DPN from a FPC Agent are outside the scope of this document.

A DPN is a logical entity that performs data-plane operations (packet movement and management). It may represent a physical DPN unit, a sub-function of a physical DPN or a collection of physical DPNs (i.e., a "virtual DPN"). A DPN may be virtual -- it may export the FPC DPN Agent interface, but be implemented as software that controls other data-plane hardware or modules that may or may not be FPC-compliant. In this document, DPNs are specified without regard for whether the implementation is virtual or physical. DPNs are connected to provide mobility management systems such as access networks, anchors and domains. The FPC agent interface enables establishment of a topology for the forwarding plane.

When a DPN is mapped to physical data-plane equipment, the FPC client can have complete knowledge of the DPN architecture, and use that information to perform DPN selection for specific sessions. On the other hand, when a virtual DPN is mapped to a collection of physical DPNs, the FPC client cannot select a specific physical DPN because it is hidden by the abstraction; only the FPC Agent can address the specific associated physical DPNs. Network architects have the flexibility to determine which DPN-selection capabilities are performed by the FPC Agent (distributed) and which by the FPC client (centralized). In this way, overlay networks can be configured without disclosing detailed knowledge of the underlying hardware to the FPC client and applications.

The abstractions in this document are designed to support many different mobility management systems and data-plane functions. The architecture and protocol design of FPC is not tied to specific types of access technologies and mobility protocols.

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 [RFC2119].

Attribute Expression: The definition of a template Property. This includes setting the type, current value, default value and if the attribute is static, i.e. can no longer be changed.

Domain: One or more DPNs that form a logical partition of network resources (e.g., a data-plane network under common network administration). A FPC client (e.g., a mobility management system) may utilize a single or multiple domains.

DPN: A data-plane node (DPN) is capable of performing data-plane features. For example, DPNs may be switches or routers, regardless of whether they are realized as hardware or purely in software.

FPC Client: A FPC Client is integrated with a mobility management system or related application, enabling control over forwarding policy, mobility sessions and DPNs via a FPC Agent.

Mobility Context: A Mobility Context contains the data-plane information necessary to efficiently send and receive traffic from a mobile node. This includes policies that are created or modified during the network’s operation - in most cases, on a per-flow or per session basis. A Mobility-Context represents the mobility sessions (or flows) which are active on a mobile node. This includes associated runtime attributes, such as tunnel endpoints, tunnel identifiers, delegated prefix(es), routing information, etc. Mobility-Contexts are associated to specific DPNs. Some pre-defined Policies may apply during mobility signaling requests. The Mobility Context supplies information about the policy settings specific to a mobile node and its flows; this information is often quite dynamic.

Mobility Session: Traffic to/from a mobile node that is expected to survive reconnection events.

Monitor: A reporting mechanism for a list of events that trigger notification messages from a FPC Agent to a FPC Client.

Policy: A Policy determines the mechanisms for managing specific traffic flows or packets. Policies specify QoS, rewriting rules for packet processing, etc. A Policy consists of one or more rules. Each rule is composed of a Descriptor and Actions. The Descriptor in a rule identifies packets (e.g., traffic flows), and the Actions apply treatments to packets that match the Descriptor in the rule. Policies can apply to Domains, DPNs, Mobile Nodes, Service-Groups, or particular Flows on a Mobile Node.

Property: An attribute-value pair for an instance of a FPC entity.

Service-Group: A set of DPN interfaces that support a specific data-plane purpose, e.g. inbound/outbound, roaming, subnetwork with common specific configuration, etc.

Template: A recipe for instantiating FPC entities. Template definitions are accessible (by name or by a key) in an indexed set. A Template is used to create specific instances (e.g., specific policies) by assigning appropriate values into the Template definition via Attribute Expression.

Template Configuration: The process by which a Template is referenced (by name or by key) and Attribute Expressions are created that change the value, default value or static nature of the Attribute, if permitted. If the Template is Extensible, new attributes MAY be added.

Tenant: An operational entity that manages mobility management systems or applications which require data-plane functions. A Tenant defines a global namespace for all entities owned by the Tenant enabling its entities to be used by multiple FPC Clients across multiple FPC Agents.

Topology: The DPNs and the links between them. For example, access nodes may be assigned to a Service-Group which peers to a Service-Group of anchor nodes.

3. FPC Design Objectives and Deployment

Using FPC, mobility control-planes and applications can configure DPNs to perform various mobility management roles as described in [I-D.ietf-dmm-deployment-models]. This fulfills the requirements described in [RFC7333].

This document defines FPC Agent and FPC Client, as well as the information models that they use. The attributes defining those models serve as the protocol elements for the interface between the FPC Agent and the FPC Client.

Mobility control-plane applications integrate features offered by the FPC Client. The FPC Client connects to FPC Agent functions. The Client and the Agent communicate based on information models described in Section 4. The models allow the control-plane to configure forwarding policies on the Agent for data-plane communications with mobile nodes.

Once the Topology of DPN(s) and domains are defined on an Agent for a data plane, the DPNs in the topology are available for further configuration. The FPC Agent connects those DPNs to manage their configurations.

A FPC Agent configures and manages its DPN(s) according to forwarding policies requested and Attributes provided by the FPC Client. Configuration commands used by the FPC agent to configure its DPN node(s) may be specific to the DPN implementation; consequently the method by which the FPC Agent carries out the specific configuration for its DPN(s) is out of scope for this document. Along with the data models, the FPC Client (on behalf of control-plane and applications) requests that the Agent configures Policies prior to the time when the DPNs start forwarding data for their mobility sessions.

This architecture is illustrated in Figure 1. A FPC Agent may be implemented in a network controller that handles multiple DPNs, or (more simply) an FPC Agent may itself be integrated into a DPN.

This document does not specify a protocol for the FPC interface; it is out of scope.

```
                           +-------------------------+
                           | Mobility Control-Plane  |
                           |          and            |
                           |      Applications       |
                           |+-----------------------+|
                           ||      FPC Client       ||
                           |+----------^------------+|
                           +-----------|-------------+
               FPC interface protocol  |
                       +---------------+-----------------+
                       |                                 |
         Network       |                                 |
         Controller    |                      DPN        |
           +-----------|-------------+        +----------|---------+
           |+----------v------------+|        |+---------v--------+|
           ||   [Data-plane model]  ||        ||[Data-plane model]||
           ||       FPC Agent       ||        ||    FPC Agent     ||
           |+-----------------------+|        |+------------------+|
           |+------------+----------+|        |                    |
           ||SB Protocol |FPC Client||        |  DPN Configuration |
           ||   Modules  |  Module  ||        +--------------------+
           |+------^-----+----^-----+|
           +-------|----------|------+
                   |          |
         Other     |          | FPC interface
        southbound |          | protocol
         protocols |          |
                   |          +-----------------+
                   |                            |
       DPN         |                 DPN        |
        +----------|---------+       +----------|---------+
        |+---------v--------+|       |+---------v--------+|
        ||  Configuration   ||       ||[Data-plane model]||
        || Protocol module  ||       ||     FPC Agent    ||
        |+------------------+|       |+------------------+|
        |                    |       |                    |
        | DPN Configuration  |       |  DPN Configuration |
        +--------------------+       +--------------------+
```

Figure 1: Reference Forwarding Policy Configuration (FPC) Architecture

The FPC architecture supports multi-tenancy; a FPC enabled data-plane supports tenants of multiple mobile operator networks and/or applications. It means that the FPC Client of each tenant connects to the FPC Agent and it MUST partition namespace and data for their data-planes. DPNs on the data-plane may fulfill multiple data-plane roles which are defined per session, domain and tenant.

Multi-tenancy permits the paritioning of data-plane entities as well as a common namespace requirement upon FPC Agents and Clients when they use the same Tenant for a common data-plane entity.

FPC information models often configuration to fit the specific needs for DPN management of a mobile node's traffic. The FPC interfaces in Figure 1 are the only interfaces required to handle runtime data in a Mobility Context. The Topology and some Policy FPC models MAY be pre-configured; in that case real-time protocol exchanges are not required for them.

The information model provides an extensibility mechanism through Templates that permits specialization for the needs of a particular vendor's equipment or future extension of the model presented in this specification.

4. FPC Mobility Information Model

The FPC information model includes the following components:

DPN Information Model,
Topology Information Model,
Policy Information Model,
Mobility-Context, and
Monitor, as illustrated in Figure 2.

```
                   :
                   |
                   +-[FPC Mobility Information Model]
                   |          |
                   |          +-[Topology Information Model]
                   |          |
                   |          +-[Policy Information Model]
                   |          |
                   |          +-[Mobility-Context]
                   |          |
                   |          +-[Monitor]
                   |
```

Figure 2: FPC Information Model structure

4.1. Model Notation and Conventions

The following conventions are used to describe the FPC information models.

Information model entities (e.g. DPNs, Rules, etc.) are defined in a hierarchical notation where all entities at the same hierarchical level are located on the same left-justified vertical position sequentially. When entities are composed of sub-entities, the sub-entities appear shifted to the right, as shown in Figure 3.

```
                              |
                              +-[entity2]
                              |         +-[entity2.1]
                              |         +-[entity2.2]
```

Figure 3: Model Notation - An Example

Some entities have one or more qualifiers placed on the right hand side of the element definition in angle-brackets. Common types include:

List: A collection of entities (some could be duplicated)

Set: A nonempty collection of entities without duplications

Name: A human-readable string

Key: A unique value. We distinguish 3 types of keys:

U-Key: A key unique across all Tenants. U-Key spaces typically involve the use of registries or language specific mechanisms that guarantee universal uniqueness of values.

G-Key: A key unique within a Tenant

L-Key: A key unique within a local namespace. For example, there may exist interfaces with the same name, e.g. "if0", in two different DPNs but there can only be one "if0" within each DPN (i.e. its local Interface-Key L-Key space).

Each entity or attribute may be optional (O) or mandatory (M). Entities that are not marked as optional are mandatory.

```
  The following example shows 3 entities:
      -- Entity1 is a globally unique key, and optionally can have
                 an associated Name
      -- Entity2 is a list
      -- Entity3 is a set and is optional
                 +
                 |
                 +-[entity1] <G-Key> (M), <Name> (O)
                 +-[entity2] <List>
                 +-[entity3] <Set> (O)
                 |
                 +
```

Figure 4

When expanding entity1 into a modeling language such as YANG it would result in two values: entity1-Key and entity1-Name.

To encourage re-use, FPC defines indexed sets of various entity Templates. Other model elements that need access to an indexed model entity contain an attribute which is always denoted as "entity-Key". When a Key attribute is encountered, the referencing model element may supply attribute values for use when the referenced entity model is instantiated. For example: Figure 5 shows 2 entities:

EntityA definition references an entityB model element.

EntityB model elements are indexed by entityB-Key.

Each EntityB model element has an entityB-Key which allows it to be uniquely identified, and a list of Attributes (or, alternatively, a Type) which specifies its form. This allows a referencing entity to create an instance by supplying entityB-Values to be inserted, in a Settings container.

```
                           .
                           .
                           |
                           +-[entityA]
                           |      +-[entityB-Key]
                           |      +-[entityB-Values]
                           .
                           .
                           |
                           +-[entityB] <L-Key> (M) <Set>
                           |      +-[entityB-Type]
                           .
                           .
```

Figure 5: Indexed sets of entities

Indexed sets are specified for each of the following kinds of entities:

Domain (See Section 4.9.3)
DPN (See Section 4.9.4)
Policy (See Section 4.9.5)
Rule (See Section 4.9.5)
Descriptor (See Figure 12)
Action (See Figure 12)
Service-Group (See Section 4.9.2, and
Mobility-Context (See Section 4.9.6)

As an example, for a Domain entity, there is a corresponding attribute denoted as "Domain-Key" whose value can be used to determine a reference to the Domain.

4.2. Templates and Attributes

In order to simplify development and maintenance of the needed policies and other objects used by FPC, the Information Models which are presented often have attributes that are not initialized with their final values. When an FPC entity is instantiated according to a template definition, specific values need to be configured for each such attribute. For instance, suppose an entity Template has an Attribute named "IPv4-Address", and also suppose that a FPC Client instantiates the entity and requests that it be installed on a DPN. An IPv4 address will be needed for the value of that Attribute before the entity can be used.

```
                  +-[Template] <U-Key, Name> (M) <Set>
                  |       +-[Attributes] <Set> (M)
                  |       +-[Extensible ~ FALSE]
                  |       +-[Entity-State ~ Initial]
                  |       +-[Version]
```

Figure 6: Template entities

Attributes: A set of Attribute names MAY be included when defining a Template for instantiating FPC entities.

Extensible: Determines whether or not entities instantiated from the Template can be extended with new non-mandatory Attributes not originally defined for the Template. Default value is FALSE. If a Template does not explicitly specify this attribute, the default value is considered to be in effect.

Entity-State: Either Initial, PartiallyConfigured, Configured, or Active. Default value is Initial. See Section 4.6 for more information about how the Entity-Status changes during the configuration steps of the Entity.

Version: Provides a version tag for the Template.

The Attributes in an Entity Template may be either mandatory or non-mandatory. Attribute values may also be associated with the attributes in the Entity Template. If supplied, the value may be either assigned with a default value that can be reconfigured later, or the value can be assigned with a static value that cannot be reconfigured later (see Section 4.3).

It is possible for a Template to provide values for all of its Attributes, so that no additional values are needed before the entity can made Active. Any instantiation from a Template MUST have at least one Attribute in order to be a useful entity unless the Template has none.

4.3. Attribute-Expressions

The syntax of the Attribute definition is formatted to make it clear. For every Attribute in the Entity Template, six possibilities are specified as follows:

'[Att-Name:]' Mandatory Attribute is defined, but template does not provide any configured value.

'[Att-Name: Att-Value]' Mandatory Attribute is defined, and has a statically configured value.

'[Att-Name: ˜ Att-Value]' Mandatory Attribute is defined, and has a default value.

'[Att-Name]' Non-mandatory Attribute may be included but template does not provide any configured value.

'[Att-Name = Att-Value]' Non-mandatory Attribute may be included and has a statically configured value.

'[Att-Name ˜ Att-Value]' Non-mandatory Attribute may be included and has a default value.

So, for example, a default value for a non-mandatory IPv4-Address attribute would be denoted by [IPv4-Address ˜ 127.0.0.1].

After a FPC Client identifies which additional Attributes have been configured to be included in an instantiated entity, those configured Attributes MUST NOT be deleted by the FPC Agent. Similarly, any statically configured value for an entity Attribute MUST NOT be changed by the FPC Agent.

Whenever there is danger of confusion, the fully qualified Attribute name MUST be used when supplying needed Attribute Values for a structured Attribute.

4.4. Attribute Value Types

For situations in which the type of an attribute value is required, the following syntax is recommended. To declare than an attribute has data type "foo", typecast the attribute name by using the parenthesized data type (foo). So, for instance, [(float) Max-Latency-in-ms:] would indicate that the mandatory Attribute "Max-Latency-in-ms" requires to be configured with a floating point value before the instantiated entity could be used. Similarly, [(float) Max-Latency-in-ms: 9.5] would statically configure a floating point value of 9.5 to the mandatory Attribute "Max-Latency-in-ms".

4.5. Namespace and Format

The identifiers and names in FPC models which reside in the same Tenant must be unique. That uniqueness must be maintained by all Clients, Agents and DPNs that support the Tenant. The Tenant namespace uniqueness MUST be applied to all elements of the tenant model, i.e. Topology, Policy and Mobility models.

When a Policy needs to be applied to Mobility-Contexts in all Tenants on an Agent, the Agent SHOULD define that policy to be visible by all Tenants. In this case, the Agent assigns a unique identifier in the Agent namespace and copies the values to each Tenant. This effectively creates a U-Key although only a G-Key is required within the Tenant.

The notation for identifiers can utilize any format with agreement between data-plane agent and client operators. The formats include but are not limited to Globally Unique IDentifiers (GUIDs), Universally Unique IDentifiers (UUIDs), Fully Qualified Domain Names (FQDNs), Fully Qualified Path Names (FQPNs) and Uniform Resource Identifiers (URIs). The FPC model does not limit the format, which could dictate the choice of FPC protocol. Nevertheless, the identifiers which are used in a Mobility model should be considered to efficiently handle runtime parameters.

4.6. Configuring Attribute Values

Attributes of Information Model components such as policy templates are configured with values as part of FPC configuration operations. There may be several such configuration operations before the template instantiation is fully configured.

Entity-Status indicates when an Entity is usable within a DPN. This permits DPN design tradeoffs amongst local storage (or other resources), over the wire request size and the speed of request processing. For example, DPN designers with constrained systems MAY only house entities whose status is Active which may result in sending over all policy information with a Mobility-Context request. Storing information elements with an entity status of "PartiallyConfigured" on the DPN requires more resources but can result in smaller over the wire FPC communication and request processing efficiency.

When the FPC Client instantiates a Policy from a Template, the Policy-Status is "Initial". When the FPC Client sends the policy to a FPC Agent for installation on a DPN, the Client often will configure appropriate attribute values for the installation, and accordingly changes the Policy-Status to "PartiallyConfigured" or "Configured". The FPC Agent will also configure Domain-specific policies and DPN-specific policies on the DPN. When configured to provide particular services for mobile nodes, the FPC Agent will apply whatever service-specific policies are needed on the DPN. When a mobile node attaches to the network data-plane within the topology under the jurisdiction of a FPC Agent, the Agent may apply policies and settings as appropriate for that mobile node. Finally, when the mobile node launches new flows, or quenches existing flows, the FPC

Agent, on behalf of the FPC Client, applies or deactivates whatever policies and attribute values are appropriate for managing the flows of the mobile node. When a "Configured" policy is de-activated, Policy-Status is changed to be "Active". When an "Active" policy is activated, Policy-Status is changed to be "Configured".

Attribute values in DPN resident Policies may be configured by the FPC Agent as follows:

Domain-Policy-Configuration: Values for Policy attributes that are required for every DPN in the domain.

DPN-Policy-Configuration: Values for Policy attributes that are required for every policy configured on this DPN.

Service-Group-Policy-Configuration: Values for Policy attributes that are required to carry out the intended Service of the Service Group.

MN-Policy-Configuration: Values for Policy attributes that are required for all traffic to/from a particular mobile node.

Service-Data-Flow-Policy-Configuration: Values for Policy attributes that are required for traffic belonging to a particular set of flows on the mobile node.

Any configuration changes MAY also supply updated values for existing default attribute values that may have been previously configured on the DPN resident policy.

Entity blocks describe the format of the policy configurations.

4.7. Entity Configuration Blocks

As described in Section 4.6, a Policy Template may be configured in several stages by configuring default or missing values for Attributes that do not already have statically configured values. A Policy-Configuration is the combination of a Policy-Key (to identify the Policy Template defining the Attributes) and the currently configured Attribute Values to be applied to the Policy Template. Policy-Configurations MAY add attributes to a Template if Extensible is True. They MAY also refine existing attributes by:

assign new values if the Attribute is not static

make attributes static if they were not

make an attribute mandatory

A Policy-Configuration MUST NOT define or refine an attribute twice. More generally, an Entity-Configuration can be defined for any configurable Indexed Set to be the combination of the Entity-Key along with a set of Attribute-Expressions that supply configuration information for the entity’s Attributes. Figure 7 shows a schematic representation for such Entity Configuration Blocks.

```
[Entity Configuration Block]
|       +-[Entity-Key] (M)
|       +-[Attribute-Expression] <Set> (M)
```

Figure 7: Entity Configuration Block

This document makes use of the following kinds of Entity Configuration Blocks:

Descriptor-Configuration

Action-Configuration

Rule-Configuration

Interface-Configuration

Service-Group-Configuration

Domain-Policy-Configuration

DPN-Policy-Configuration

Policy-Configuration

MN-Policy-Configuration

Service-Data-Flow-Policy-Configuration

4.8. Information Model Checkpoint

The Information Model Checkpoint permits Clients and Tenants with common scopes, referred to in this specification as Checkpoint BaseNames, to track the state of provisioned information on an Agent. The Agent records the Checkpoint BaseName and Checkpoint value set by a Client. When a Client attaches to the Agent it can query to determine the amount of work that must be executed to configure the Agent to a specific BaseName / checkpoint revision.

Checkpoints are defined for the following information model components:

Service-Group

DPN Information Model

Domain Information Model

Policy Information Model

4.9. Information Model Components

4.9.1. Topology Information Model

The Topology structure specifies DPNs and the communication paths between them. A network management system can use the Topology to select the most appropriate DPN resources for handling specific session flows.

The Topology structure is illustrated in Figure 8 (for definitions see Section 2):

```
                         |
                         +-[Topology Information Model]
                         |          +-[Extensible: FALSE]
                         |          +-[Service-Group]
                         |          +-[DPN] <Set>
                         |          +-[Domain] <Set>
```

Figure 8: Topology Structure

4.9.2. Service-Group

Service-Group-Set is collection of DPN interfaces serving some data-plane purpose including but not limited to DPN Interface selection to fulfill a Mobility-Context. Each Group contains a list of DPNs (referenced by DPN-Key) and selected interfaces (referenced by Interface-Key). The Interfaces are listed explicitly (rather than referred implicitly by its specific DPN) so that every Interface of a DPN is not required to be part of a Group. The information provided is sufficient to ensure that the Protocol, Settings (stored in the Service-Group-Configuration) and Features relevant to successful interface selection is present in the model.

```
   |
   +-[Service-Group] <G-Key>, <Name> (O) <Set>
   |           +-[Extensible: FALSE]
   |           +-[Role] <U-Key>
   |           +-[Protocol] <Set>
   |           +-[Feature] <Set> (O)
   |           +-[Service-Group-Configuration] <Set> (O)
   |           +-[DPN-Key] <Set>
   |           |           +-[Referenced-Interface] <Set>
   |           |           |       +-[Interface-Key] <L-Key>
   |           |           |       +-[Peer-Service-Group-Key] <Set> (O)
```

Figure 9: Service Group

Each Service-Group element contains the following information:

Service-Group-Key: A unique ID of the Service-Group.

Service-Group-Name: A human-readable display string.

Role: The role (MAG, LMA, etc.) of the device hosting the interfaces of the DPN Group.

Protocol-Set: The set of protocols supported by this interface (e.g., PMIP, S5-GTP, S5-PMIP etc.). The protocol MAY be only its name, e.g. 'gtp', but many protocols implement specific message sets, e.g. s5-pmip, s8-pmip. When the Service-Group supports specific protocol message sub-subsets the Protocol value MUST include this information.

Feature-Set: An optional set of static features which further determine the suitability of the interface to the desired operation.

Service-Group-Configuration-Set: An optional set of configurations that further determine the suitability of an interface for the specific request. For example: SequenceNumber=ON/OFF.

DPN-Key-Set: A key used to identify the DPN.

Referenced-Interface-Set: The DPN Interfaces and peer Service-Groups associated with them. Each entry contains

Interface-Key: A key that is used together with the DPN-Key, to create a key that is refers to a specific DPN interface definition.

Peer-Service-Group-Key: Enables location of the peer Service-Group for this Interface.

4.9.3. Domain Information Model

A Domain-Set represents a group of heterogeneous Topology resources typically sharing a common administrative authority. Other models, outside of the scope of this specification, provide the details for the Domain.

```
             |
             +-[Domain] <G-Key>, <Name> (O) <Set>
             |       +-[Domain-Policy-Configuration] (O) <Set>
             |
```

Figure 10: Domain Information Model

Each Domain entry contains the following information:

Domain-Key: Identifies and enables reference to the Domain.

Domain-Name: A human-readable display string naming the Domain.

4.9.4. DPN Information Model

A DPN-Set contains some or all of the DPNs in the Tenant's network. Some of the DPNs in the Set may be identical in functionality and only differ by their Key.

```
               |
               +-[DPN] <G-Key>, <Name> (O) <Set>
               |     +-[Extensible: FALSE]
               |     +-[Interface] <L-Key> <Set>
               |     |      +-[Role] <U-Key>
               |     |      +-[Protocol] <Set>
               |     |      +-[Interface-Configuration] <Set> (O)
               |     +-[Domain-Key]
               |     +-[Service-Group-Key] <Set> (O)
               |     +-[DPN-Policy-Configuration] <List> (M)
               |     +-[DPN-Resource-Mapping-Reference] (O)
```

Figure 11: DPN Information Model

Each DPN entry contains the following information:

DPN-Key: A unique Identifier of the DPN.

DPN-Name: A human-readable display string.

Domain-Key: A Key providing access to the Domain information about the Domain in which the DPN resides.

Interface-Set: The Interface-Set references all interfaces (through which data packets are received and transmitted) available on the DPN. Each Interface makes use of attribute values that are specific to that interface, for example, the MTU size. These do not affect the DPN selection of active or enabled interfaces. Interfaces contain the following information:

Role: The role (MAG, LMA, PGW, AMF, etc.) of the DPN.

Protocol (Set): The set of protocols supported by this interface (e.g., PMIP, S5-GTP, S5-PMIP etc.). The protocol MAY implement specific message sets, e.g. s5-pmip, s8-pmip. When a protocol implements such message sub-subsets the Protocol value MUST include this information.

Interface-Configuration-Set: Configurable settings that further determine the suitability of an interface for the specific request. For example: SequenceNumber=ON/OFF.

Service-Group-Set: The Service-Group-Set references all of the Service-Groups which have been configured using Interfaces hosted on this DPN. The purpose of a Service-Group is not to describe each interface of each DPN, but rather to indicate interface types for use during the DPN selection process, when a DPN with specific interface capabilities is required.

DPN-Policy-Configuration: A list of Policies that have been configured on this DPN. Some may have values for all attributes, and some may require further configuration. Each Policy-Configuration has a key to enable reference to its Policy-Template. Each Policy-Configuration also has been configured to supply missing and non-default values to the desired Attributes defined within the Policy-Template.

DPN-Resource-Mapping-Reference (O): A reference to the underlying implementation, e.g. physical node, software module, etc. that supports this DPN. Further specification of this attribute is out of scope for this document.

4.9.5. Policy Information Model

The Policy Information Model defines and identifies Rules for enforcement at DPNs. A Policy is basically a set of Rules that are to be applied to each incoming or outgoing packet at a DPN interface. Rules comprise Descriptors and a set of Actions. The Descriptors, when evaluated, determine whether or not a set of Actions will be performed on the packet. The Policy structure is independent of a policy context.

In addition to the Policy structure, the Information Model (per Section 4.9.6) defines Mobility-Context. Each Mobility-Context may be configured with appropriate Attribute values, for example depending on the identity of a mobile node.

Traffic descriptions are defined in Descriptors, and treatments are defined separately in Actions. A Rule-Set binds Descriptors and associated Actions by reference, using Descriptor-Key and Action-Key. A Rule-Set is bound to a policy in the Policy-Set (using Policy-Key), and the Policy references the Rule definitions (using Rule-Key).

```
              |
              +-[Policy Information Model]
              |      +-[Extensible:]
              |      +-[Policy-Template] <G-Key> (M) <Set>
              |      |       +-[Policy-Configuration] <Set> (O)
              |      |       +-[Rule-Template-Key] <List> (M)
              |      |       |       +-[Precedence] (M)
              |      +-[Rule-Template] <L-Key> (M) <Set>
              |      |       +-[Descriptor-Match-Type] (M)
              |      |       +-[Descriptor-Configuration] <Set> (M)
              |      |       |       +-[Direction] (O)
              |      |       +-[Action-Configuration] <Set> (M)
              |      |       |       +-[Action-Order] (M)
              |      |       +-[Rule-Configuration] (O)
              |      +-[Descriptor-Template] <L-Key> (M) <Set>
              |      |       +-[Descriptor-Type] (O)
              |      |       +-[Attribute-Expression] <Set> (M)
              |      +-[Action-Template] <L-Key> (M) <Set>
              |              +-[Action-Type] (O)
              |      |       +-[Attribute-Expression] <Set> (M)
```

Figure 12: Policy Information Model

The Policy structure defines Policy-Set, Rule-Set, Descriptor-Set, and Action-Set, as follows:

Policy-Template: <Set> A set of Policy structures, indexed by Policy-Key, each of which is determined by a list of Rules referenced by their Rule-Key. Each Policy structure contains the following:

Policy-Key: Identifies and enables reference to this Policy definition.

Rule-Template-Key: Enables reference to a Rule template definition.

Rule-Precedence: For each Rule identified by a Rule-Template-Key in the Policy, specifies the order in which that Rule must be applied. The lower the numerical value of Precedence, the higher the rule precedence. Rules with equal precedence MAY be executed in parallel if supported by the DPN. If this value is absent, the rules SHOULD be applied in the order in which they appear in the Policy.

Rule-Template-Set: A set of Rule Template definitions indexed by Rule-Key. Each Rule is defined by a list of Descriptors (located by Descriptor-Key) and a list of Actions (located by Action-Key) as follows:

Rule-Template-Key: Identifies and enables reference to this Rule definition.

Descriptor-Match-Type Indicates whether the evaluation of the Rule proceeds by using conditional-AND, or conditional-OR, on the list of Descriptors.

Descriptor-Configuration: References a Descriptor template definition, along with an expression which names the Attributes for this instantiation from the Descriptor-Template and also specifies whether each Attribute of the Descriptor has a default value or a statically configured value, according to the syntax specified in Section 4.2.

Direction: Indicates if a rule applies to uplink traffic, to downlink traffic, or to both uplink and downlink traffic. Applying a rule to both uplink and downlink traffic, in case of symmetric rules, eliminates the requirement for a separate entry for each direction. When not present, the direction is implied by the Descriptor’s values.

Action-Configuration: References an Action Template definition, along with an expression which names the Attributes for this instantiation from the Action-Template and also specifies whether each Attribute of the Action has a default value or a statically configured value, according to the syntax specified in Section 4.2.

Action-Order: Defines the order in which actions are executed when the associated traffic descriptor selects the packet.

Descriptor-Template-Set: A set of traffic Descriptor Templates, each of which can be evaluated on the incoming or outgoing packet, returning a TRUE or FALSE value, defined as follows:

Descriptor-Template-Key: Identifies and enables reference to this descriptor template definition.

Attribute-Expression: An expression which defines an Attribute in the Descriptor-Template and also specifies whether the Template also defines a default value or a statically configured value for the Attribute of the Descriptor has, according to the syntax specified in Section 4.2.

Descriptor-Type: Identifies the type of descriptor, e.g. an IPv6 traffic selector per [RFC6088].

Action-Template-Set: A set of Action Templates defined as follows:

Action-Template-Key: Identifies and enables reference to this action template definition.

Attribute-Expression: An expression which defines an Attribute in the Action-Template and also specifies whether the Template also defines a default value or a statically configured value for the Attribute of the Action has, according to the syntax specified in Section 4.2.

Action-Type: Identifies the type of an action for unambiguous interpretation of an Action-Value entry.

4.9.6. Mobility-Context Information Model

The Mobility-Context structure holds entries associated with a mobile node and its mobility sessions (flows). It is created on a DPN during the mobile node's registration to manage the mobile node's flows. Flow information is added or deleted from the Mobility-Context as needed to support new flows or to deallocate resources for flows that are deactivated. Descriptors are used to characterize the nature and resource requirement for each flow.

Termination of a Mobility-Context implies termination of all flows represented in the Mobility-Context, e.g. after deregistration of a mobile node. If any Child-Contexts are defined, they are also terminated.

```
     +-[Mobility-Context] <G-Key> <Set>
     |            +-[Extensible:~ FALSE]
     |            +-[Delegating-IP-Prefix:] <Set> (O)
     |            +-[Parent-Context] (O)
     |            +-[Child-Context] <Set> (O)
     |            +-[Service-Group-Key] <Set> (O)
     |            +-[Mobile-Node]
     |            |       +-[IP-Address] <Set> (O))
     |            |       +-[MN-Policy-Configuration] <Set>
     |            +-[Domain-Key]
     |            |       +-[Domain-Policy-Configuration] <Set>
     |            +-[DPN-Key] <Set>
     |            |       +-[Role]
     |            |       +-[DPN-Policy-Configuration] <Set>
     |            |       +-[ServiceDataFlow] <L-Key> <Set> (O)
     |            |       |       +-[Service-Group-Key] (O)
     |            |       |       +-[Interface-Key] <Set>
     |            |       |       +-[ServiceDataFlow-Policy-
                                        Configuration] <Set> (O)
     |            |       |       |       +-[Direction]
```

Figure 13: Mobility-Context Information Model

The Mobility-Context Substructure holds the following entries:

Mobility-Context-Key: Identifies a Mobility-Context

Delegating-IP-Prefix-Set: Delegated IP Prefixes assigned to the Mobility-Context

Parent-Context: If present, a Mobility Context from which the Attributes and Attribute Values of this Mobility Context are inherited.

Child-Context-Set: A set of Mobility Contexts which inherit the Attributes and Attribute Values of this Mobility Context.

Service-Group-Key: Service-Group(s) used during DPN assignment and re-assignment.

Mobile-Node: Attributes specific to the Mobile Node. It contains the following

IP-Address-Set IP addresses assigned to the Mobile Node.

MN-Policy-Configuration-Set For each MN-Policy in the set, a key and relevant information for the Policy Attributes.

Domain-Key: Enables access to a Domain instance.

Domain-Policy-Configuration-Set: For each Domain-Policy in the set, a key and relevant information for the Policy Attributes.

DPN-Key-Set: Enables access to a DPN instance assigned to a specific role, i.e. this is a Set that uses DPN-Key and Role as a compound key to access specific set instances.

Role: Role this DPN fulfills in the Mobility-Context.

DPN-Policy-Configuration-Set: For each DPN-Policy in the set, a key and relevant information for the Policy Attributes.

ServiceDataFlow-Key-Set: Characterizes a traffic flow that has been configured (and provided resources) on the DPN to support data-plane traffic to and from the mobile device.

Service-Group-Key: Enables access to a Service-Group instance.

Interface-Key-Set: Assigns the selected interface of the DPN.

ServiceDataFlow-Policy-Configuration-Set: For each Policy in the set, a key and relevant information for the Policy Attributes.

Direction: Indicates if the reference Policy applies to uplink or downlink traffic, or to both, uplink- and downlink traffic. Applying a rule to both, uplink- and downlink traffic, in case of symmetric rules, allows omitting a separate entry for each direction. When not present the value is assumed to apply to both directions.

4.9.7. Monitor Information Model

Monitors provide a mechanism to produce reports when events occur. A Monitor will have a target that specifies what is to be watched.

The attribute/entity to be monitored places certain constraints on the configuration that can be specified. For example, a Monitor using a Threshold configuration cannot be applied to a Mobility-Context, because it does not have a threshold. Such a monitor configuration could be applied to a numeric threshold property of a Context.

```
                           |
                           +-[Monitor] <G-Key> <List>
                           |         +-[Extensible:]
                           |         +-[Target:]
                           |         +-[Deferrable]
                           |         +-[Configuration]
```

Figure 14: Monitor Substructure

Monitor-Key: Identifies the Monitor.

Target: Description of what is to be monitored. This can be a Service Data Flow, a Policy installed upon a DPN, values of a Mobility-Context, etc. The target name is the absolute information model path (separated by '/') to the attribute / entity to be monitored.

Deferrable: Indicates that a monitoring report can be delayed up to a defined maximum delay, set in the Agent, for possible bundling with other reports.

Configuration: Determined by the Monitor subtype. The monitor report is specified by the Configuration. Four report types are defined:

* "Periodic" reporting specifies an interval by which a notification is sent.

* "Event-List" reporting specifies a list of event types that, if they occur and are related to the monitored attribute, will result in sending a notification.

* "Scheduled" reporting specifies the time (in seconds since Jan 1, 1970) when a notification for the monitor should be sent. Once this Monitor's notification is completed the Monitor is automatically de-registered.

* "Threshold" reporting specifies one or both of a low and high threshold. When these values are crossed a corresponding notification is sent.

5. Security Considerations

Detailed protocol implementations for DMM Forwarding Policy Configuration must ensure integrity of the information exchanged between a FPC Client and a FPC Agent. Required Security Associations may be derived from co-located functions, which utilize the FPC Client and FPC Agent respectively.

General usage of FPC MUST consider the following:

FPC Naming Section 4.5 permits arbitrary string values but a user MUST avoid placing sensitive or vulnerable information in those values.

Policies that are very narrow and permit the identification of specific traffic, e.g. that of a single user, SHOULD be avoided.

6. IANA Considerations

TBD

7. Work Team Participants

Participants in the FPSM work team discussion include Satoru Matsushima, Danny Moses, Sri Gundavelli, Marco Liebsch, Pierrick Seite, Alper Yegin, Carlos Bernardos, Charles Perkins and Fred Templin.

8. References

8.1. 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>.

[RFC6088] Tsirtsis, G., Giarreta, G., Soliman, H., and N. Montavont, "Traffic Selectors for Flow Bindings", RFC 6088, DOI 10.17487/RFC6088, January 2011, <https://www.rfc-editor.org/info/rfc6088>.

8.2. Informative References

[I-D.bertz-dime-policygroups]
Bertz, L. and M. Bales, "Diameter Policy Groups and Sets", Work in Progress, Internet-Draft, draft-bertz-dime-policygroups-06, 18 June 2018, <http://www.ietf.org/internet-drafts/draft-bertz-dime-policygroups-06.txt>.

[I-D.ietf-dmm-deployment-models]
Gundavelli, S. and S. Jeon, "DMM Deployment Models and Architectural Considerations", Work in Progress, Internet-Draft, draft-ietf-dmm-deployment-models-04, 15 May 2018, <http://www.ietf.org/internet-drafts/draft-ietf-dmm-deployment-models-04.txt>.

[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>.

Appendix A. Implementation Status

Three FPC Agent implementations have been made to date. The first was based upon Version 03 of the draft and followed Model 1. The second follows Version 04 of the document. Both implementations were OpenDaylight plug-ins developed in Java by Sprint. Version 04 is now primarily enhanced by GS Labs. Version 03 was known as fpcagent and version 04’s implementation is simply referred to as ’fpc’. A third has been developed on an ONOS Controller for use in MCORD projects.

fpcagent’s intent was to provide a proof of concept for FPC Version 03 Model 1 in January 2016 and research various errors, corrections and optimizations that the Agent could make when supporting multiple DPNs.

As the code developed to support OpenFlow and a proprietary DPN from a 3rd party, several of the advantages of a multi-DPN Agent became obvious including the use of machine learning to reduce the number of Flows and Policy entities placed on the DPN. This work has driven new efforts in the DIME WG, namely Diameter Policy Groups [I-D.bertz-dime-policygroups].

A throughput performance of tens per second using various NetConf based solutions in OpenDaylight made fpcagent, based on version 03, undesirable for call processing. The RPC implementation improved throughput by an order of magnitude but was not useful based upon FPC’s Version 03 design using two information models. During this time the features of version 04 and its converged model became attractive and the fpcagent project was closed in August 2016. fpcagent will no longer be developed and will remain a proprietary implementation.

The learnings of fpcagent has influenced the second project, fpc. Fpc is also an OpenDaylight project but is an open source release as the Opendaylight FpcAgent plugin (https://wiki.opendaylight.org/view/Project_Proposals:FpcAgent). This project is scoped to be a fully compliant FPC Agent that supports multiple DPNs including those that communicate via OpenFlow. The following features present in this draft and others developed by the FPC development team have already led to an order of magnitude improvement.

Migration of non-realtime provisioning of entities such as topology and policy allowed the implementation to focus only on the rpc.

Using only 5 messages and 2 notifications has also reduced implementation time.

Command Sets, an optional feature in this specification, have eliminated 80% of the time spent determining what needs to be done with a Context during a Create or Update operation.

Op Reference is an optional feature modeled after video delivery. It has reduced unnecessary cache lookups. It also has the additional benefit of allowing an Agent to become cacheless and effectively act as a FPC protocol adapter remotely with multi-DPN support or co-located on the DPN in a single-DPN support model.

Multi-tenant support allows for Cache searches to be partitioned for clustering and performance improvements. This has not been capitalized upon by the current implementation but is part of the development roadmap.

Use of Contexts to pre-provision policy has also eliminated any processing of Ports for DPNs which permitted the code for CONFIGURE and CONF_BUNDLE to be implemented as a simple nested FOR loops (see below).

Initial v04 performance results without code optimizations or tuning allow reliable provisioning of 1K FPC Mobility-Contexts processed per second on a 12 core server. This results in 2x the number of transactions on the southbound interface to a proprietary DPN API on the same machine.

fpc currently supports the following:

1 proprietary DPN API

Policy and Topology as defined in this specification using OpenDaylight North Bound Interfaces such as NetConf and RestConf

CONFIG and CONF_BUNDLE (all operations)

DPN assignment, Tunnel allocations and IPv4 address assignment by the Agent or Client.

Immediate Response is always an OK_NOTIFY_FOLLOWS.

```
assignment system (receives rpc call):
  perform basic operation integrity check
  if CONFIG then
    goto assignments
    if assignments was ok then
      send request to activation system
      respond back to client with assignment data
    else
      send back error
    end if
  else if CONF_BUNDLE then
    for each operation in bundles
    goto assignments
    if assignments was ok then
      hold onto data
    else
      return error with the assignments that occurred in
        prior operations (best effort)
    end if
    end for
    send bundles to activation systems
  end if

assignments:
  assign DPN, IPv4 Address and/or tunnel info as required
  if an error occurs undo all assignments in this operation
  return result

activation system:
  build cache according to op-ref and operation type
  for each operation
    for each Context
      for each DPN / direction in Context
        perform actions on DPN according to Command Set
      end for
    end for
  end for
  commit changes to in memory cache
  log transaction for tracking and notification
                                (CONFIG_RESULT_NOTIFY)
```

Figure 15: fpc pseudo code

For further information please contact Lyle Bertz who is also a co-author of this document.

NOTE: Tenant support requires binding a Client ID to a Tenant ID (it is a one to many relation) but that is outside of the scope of this specification. Otherwise, the specification is complete in terms of providing sufficient information to implement an Agent.

Authors' Addresses

Satoru Matsushima
SoftBank
1-9-1,Higashi-Shimbashi,Minato-Ku,
Japan

Email: satoru.matsushima@g.softbank.co.jp

Lyle Bertz
6220 Sprint Parkway
Overland Park KS, 66251,
United States of America

Email: lylebe551144@gmail.com

Marco Liebsch
NEC Laboratories Europe
NEC Europe Ltd.
Kurfuersten-Anlage 36
D-69115 Heidelberg
Germany

Phone: +49 6221 4342146
Email: liebsch@neclab.eu

Sri Gundavelli
Cisco
170 West Tasman Drive
San Jose, CA 95134
United States of America

Email: sgundave@cisco.com

Danny Moses

Email: danny.moses@intel.com

Charles E. Perkins
Futurewei Inc.
2330 Central Expressway
Santa Clara, CA 95050
United States of America

Phone: +1-408-330-4586
Email: charliep@computer.org

DMM Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 2, 2017

Z. Yan
CNNIC
J. Lee
Sangmyung University
X. Lee
CNNIC

July 1, 2016

Home Network Prefix Renumbering in PMIPv6
draft-ietf-dmm-hnprenum-03

Abstract

In the basic Proxy Mobile IPv6 (PMIPv6) specification, a Mobile Node (MN) is assigned with a Home Network Prefix (HNP) during its initial attachment and the MN configures its Home Address (HoA) with the HNP. During the movement of the MN, the HNP is remained unchanged to keep ongoing communications associated with the HoA. However, the current PMIPv6 specification does not specify related operations when an HNP renumbering is happened. In this document, a solution to support the HNP renumbering is proposed, as an update of the PMIPv6 specification.

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 [RFC2119]

Status of This Memo

This Internet-Draft will expire on January 2, 2017.

Copyright Notice

Table of Contents

1. Introduction

Network managers currently prefer Provider Independent (PI) addressing for IPv6 to attempt to minimize the need for future possible renumbering. However, a widespread use of PI addresses may cause Border Gateway Protocol (BGP) scaling problems. It is thus desirable to develop tools and practices that make IPv6 renumbering a simpler process to reduce demand for IPv6 PI space [RFC6879]. In this document, we aim to solve the HNP renumbering problem when the HNP in PMIPv6 [RFC5213] is not the type of PI.

2. Usage Scenarios

There are a number of reasons why the HNP renumbering support in PMIPv6 is useful and some scenarios are identified below:

o Scenario 1: the HNP set used by a PMIPv6 service provider is assigned by a different Internet Service Provider (ISP), and then the HNP renumbering may happen if the PMIPv6 service provider switches to a different ISP.

- Scenario 2: multiple Local Mobility Anchors (LMAs) may be deployed by the same PMIPv6 service provider, and then each LMA may serve for a specific HNP set. In this case, the HNP of an MN may change if the current serving LMA switches to another LMA but without inheriting the assigned HNP set [RFC6463].

- Scenario 3: the PMIPv6 HNP renumbering may be caused by the re-building of the network architecture as the companies split, merge, grow, relocate, or reorganize. For example, the PMIPv6 service provider may reorganize its network topology.

In the scenario 1, we assume that only the HNP is renumbered while the serving LMA remains unchanged and this is the basic scenario considered in this document. In the scenario 2 and scenario 3, more complex results may be caused, for example, the HNP renumbering may happen due to the switchover of a serving LMA.

In the Mobile IPv6 (MIPv6) protocol, when a home network prefix changes, the Home Agent (HA) will actively notify the new prefix to its MN and then the renumbering of the Home Network Address (HoA) can be well supported [RFC6275]. In the basic PMIPv6, the PMIPv6 binding is triggered by a Mobile Access Gateway (MAG), which detects the attachment of the MN. A scheme is also needed for the LMA to immediately initiate the PMIPv6 binding state refreshment during the HNP renumbering process. Although this issue is also mentioned in Section 6.12 of [RFC5213], the related solution has not been specified.

3. PMIPv6 Extensions

When the HNP renumbering happens in PMIPv6, the LMA has to notify a new HNP to an MAG and then the MAG has to announce the new HNP to the attached MN accordingly. Also, the LMA and the MAG must update the routing states for the HNP and the related addresses. To support this procedure, [RFC7077] can be adopted which specifies an asynchronous update from the LMA to the MAG about specific session parameters. This document considers the following two cases:

(1) HNP is renumbered under the same LMA

In this case, the LMA remains unchanged as in the scenario 1 and scenario 3. The operation steps are shown in Figure 1.

```
   +-----+                +-----+                +-----+
   | MN  |                | MAG |                | LMA |
   +-----+                +-----+                +-----+
     |                      |                      |
     |                      |           Allocate new HNP
     |                      |                      |
     |                      |<------------- UPN ---|
     |                      |                      |
     |                      |                      |
     |                      |                      |
     |<-----RA/DHCP --------|                      |
     |                      |                      |
   Address configuration    |                      |
     |                      |                      |
     |              Update binding&routing states  |
     |                      |                      |
     |                      |--- UPA ------------->|
     |                      |                      |
     |                      |     Update binding&routing states
     |                      |                      |
```

Figure 1: Signaling call flow of the HNP renumbering

- When a PMIPv6 service provider renumbers the HNP set under the same LMA, the serving LMA will initiate the HNP renumbering operation. The LMA allocates a new HNP for the related MN.

- The LMA sends the Update Notification (UPN) message to the MAG to update the HNP information. If the Dynamic Host Configuration Protocol (DHCP) is used to allocate the address, the new HNP should be also notified to the DHCP infrastructure.

- Once the MAG receives this UPN message, it recognizes that the related MN has the new HNP. Then the MAG should notify the MN about the new HNP with a Router Advertisement (RA) message or allocate a new address within the new HNP through a DHCP procedure.

- After the MN obtains the HNP information through the RA message, it deletes the old HoA and configures a new HoA with the newly allocated HNP.

- When the new HNP is announced or the new address is configured to the MN successfully, the MAG updates the related binding and routing states. Then the MAG sends back the Update Notification Acknowledgement (UPA) message to the LMA for the notification of successful update of the HNP, related binding state, and routing state. Then the LMA updates the routing and binding information corresponding to the MN to replace the old HNP with the new one.

(2) HNP renumbering caused by the LMA switchover

Since the HNP is assigned by the LMA, the HNP renumbering may be caused by the LMA switchover, as in the scenario 2 and scenario 3.

The information of LMA is the basic configuration information of MAG. When the LMA changes, the related profile should be updated by the service provider. In this way, the MAG initiates the registration to the new LMA as specified in [RFC5213]. When the HNP renumbering is caused in this case, the new HNP information is sent by the LMA during the new binding procedure. Accordingly, the MAG withdraws the old HNP of the MN and announces the new HNP to the MN as like the case of the HNP is renumbered under the same LMA.

4. Session Connectivity

The HNP renumbering may cause the disconnection of the ongoing communications of the MN. Basically, there are two modes to manage the session connectivity during the HNP renumbering.

(1) Soft-mode

The LMA will temporarily maintain the state of the old HNP during the HNP renumbering (after the UPA reception) in order to redirect the packets to the MN before the MN reconnects the ongoing session and notifies its new HoA to the Correspondent Node (CN). This mode is aiming to reduce the packet loss during the HNP renumbering but the binding state corresponding to the old HNP should be marked for example as transient binding [RFC6058]. This temporary binding should only be used for the downwards packet transmission and the LMA should stop broadcasting the routing information about the old HNP if the old HNP is no longer anchored at this LMA.

(2) Hard-mode

If the HNP renumbering happens with the switchover of the LMA, the hard-mode is recommended to keep the protocol simple. In this mode, the LMA deletes the binding state of the old HNP after it receives the UPA message from the MAG and the LMA silently discards the packets destined to the old HNP.

5. Message Format

(1) UPN message

In the UPN message sent from the LMA to the MAG, the notification reason is set to 2 (UPDATE-SESSION-PARAMETERS). Besides, the HNP Option [RFC5213] containing the new HNP and the Mobile Node Identifier Option [RFC4283] carrying identifier of MN are contained as Mobility Options of UPN. The order of HNP Option and Mobile Node Identifier Option in the UPN message is not mandated in this draft.

(2) UPA message

The MAG sends this message in order to acknowledge that it has received an UPN message with the (A) flag set and to indicate the status after processing the message. When the MAG did not successfully renumber the HNP which is required in the UPN message, the Status Code of 128 is set in the UPA message and the following operation of LMA is PMIPv6 service provider specific.

(3) RA Message

When the RA message is used by the MAG to advise the new HNP, two Prefix Information Options are contained in the RA message [RFC4861]. In the first Prefix Information Option, the old HNP is carried but both the related Valid Lifetime and Preferred Lifetime are set to 0. In the second Prefix Information Option, the new HNP is carried with the Valid Lifetime and Preferred Lifetime set to larger than 0.

(4) DHCP Message

When the DHCP is used in PMIPv6 to configure the addresses for the MN, new IPv6 address(es) (e.g., HoA) will be generated based on the new HNP and the related DHCP procedure is also triggered by the reception of UPN message [RFC3315].

6. Other Issues

In order to maintain the reachability of the MN, the Domain Name System (DNS) resource record corresponding to this MN may need to be updated when the HNP of MN changes [RFC3007]. However, this is beyond the scope of this document.

7. Security Considerations

The protection of UPN and UPA messages in this document follows [RFC5213] and [RFC7077]. This extension causes no further security problem.

8. IANA Considerations

This document presents no IANA considerations.

9. References

9.1. 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, <http://www.rfc-editor.org/info/rfc2119>.

[RFC3007] Wellington, B., "Secure Domain Name System (DNS) Dynamic Update", RFC 3007, DOI 10.17487/RFC3007, November 2000, <http://www.rfc-editor.org/info/rfc3007>.

[RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July 2003, <http://www.rfc-editor.org/info/rfc3315>.

[RFC4283] Patel, A., Leung, K., Khalil, M., Akhtar, H., and K. Chowdhury, "Mobile Node Identifier Option for Mobile IPv6 (MIPv6)", RFC 4283, DOI 10.17487/RFC4283, November 2005, <http://www.rfc-editor.org/info/rfc4283>.

[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, <http://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, <http://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, <http://www.rfc-editor.org/info/rfc6275>.

[RFC6463] Korhonen, J., Ed., Gundavelli, S., Yokota, H., and X. Cui, "Runtime Local Mobility Anchor (LMA) Assignment Support for Proxy Mobile IPv6", RFC 6463, DOI 10.17487/RFC6463, February 2012, <http://www.rfc-editor.org/info/rfc6463>.

[RFC7077] Krishnan, S., Gundavelli, S., Liebsch, M., Yokota, H., and J. Korhonen, "Update Notifications for Proxy Mobile IPv6", RFC 7077, DOI 10.17487/RFC7077, November 2013, <http://www.rfc-editor.org/info/rfc7077>.

9.2. Informative References

[RFC6058] Liebsch, M., Ed., Muhanna, A., and O. Blume, "Transient Binding for Proxy Mobile IPv6", RFC 6058, DOI 10.17487/RFC6058, March 2011, <http://www.rfc-editor.org/info/rfc6058>.

[RFC6879] Jiang, S., Liu, B., and B. Carpenter, "IPv6 Enterprise Network Renumbering Scenarios, Considerations, and Methods", RFC 6879, DOI 10.17487/RFC6879, February 2013, <http://www.rfc-editor.org/info/rfc6879>.

Authors’ Addresses

Zhiwei Yan
CNNIC
No.4 South 4th Street, Zhongguancun
Beijing 100190
China

EMail: yan@cnnic.cn

Jong-Hyouk Lee
Sangmyung University
31, Sangmyeongdae-gil, Dongnam-gu
Cheonan 31066
Republic of Korea

EMail: jonghyouk@smu.ac.kr

Xiaodong Lee
CNNIC
No.4 South 4th Street, Zhongguancun
Beijing 100190
China

EMail: xl@cnnic.cn

DMM WG
Internet-Draft
Intended status: Standards Track
Expires: January 2, 2017

D. Patki
S. Gundavelli
Cisco
J. Lee
Sangmyung University
Q. Fu
China Mobile
L. Bertz
Sprint

July 1, 2016

LMA Controlled MAG Session Parameters
draft-ietf-dmm-lma-controlled-mag-params-02.txt

Abstract

This specification defines a new extension, LMA-Controlled-MAG-Session-Params to Proxy Mobile IPv6. This option can be used by the LMA in PMIPv6 signaling for notifying the MAG to conform to various parameters contained in this extension.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

A large PMIPv6 deployment, such as residential deployment, can have tens of thousands of MAGs spread across geographical locations. While it can be operationally challenging to manage such a large number of MAGs, it can also be very difficult to ensure configuration consistency across all the MAGs if they are not centrally managed. Configuring aggressive values of parameters such as re-registration timeout and heartbeat interval can potentially create considerable signaling load on the LMA. This document provides a new option to enable the LMA to control various parameters on the MAG such as the re-registration frequency [RFC5213] and heartbeat frequency [RFC5847]. With this option, the configuration of these tunable parameters done centrally on the LMA enables Service Providers to have better control on the behavior of the MAGs with deterministic singaling load on the LMA.

2. Conventions and Terminology

2.1. Conventions

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 [RFC2119].

2.2. Terminology

All the terms used in this document are to be interpreted as defined in [RFC5213], [RFC5847] and [RFC7563].

3. Protocol Extension

The LMA Controlled MAG Parameters (LCMP) option is a mobility header option used to exchange information related to the parameters that a local mobility anchor enforces on a mobile access gateway. The option can be included in Proxy Binding Acknowledgement (PBA) message only, and there MUST NOT be more than a single instance of this mobility option in a mobility message. This mobility option MUST contain one or more LMA Controlled MAG Parameters sub-options. The suboptions are defined in Section 3.1. The alignment of this option MUST be 4n [RFC2460].

```
    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      |          Reserved             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                  ... LCMP Sub-option(s) ...                   ~
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: LMA Controlled MAG Parameters Option

Type

MUST be set to the value of IANA-1, indicating that it is a LMA-Controlled-MAG-Parameters option.

Length

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

Reserved

MUST be set to zero when sending and ignored when received.

LCMP Sub-option(s)

LCMP Sub-options are described in the below sections. The sub-options are optional and can be present in any order.

3.1. Format of the LCMP Sub-Options

The LMA Controlled MAG Parameters sub-options are used for carrying information elements related to various parameters that need to be configured on the MAG. These sub-options can be included in the LMA Controlled MAG Parameters option defined in Section 3. The alignment of the sub-option MUST be 4n. The format of this sub-option is as follows.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   LCMP Type   | LCMP Length   |       Sub-Option Data         ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: LMA Controlled MAG Parameters Sub-Option

Type

8-bit unsigned integer indicating the type of the LMA Controlled MAG Parameters sub-option. This specification defines the following types:

0 - Reserved

1 - Binding Refresh Control Sub-Option

2 - Heartbeat Control Sub-Option

Length

8-bit unsigned integer indicating the number of octets needed to encode the Option Data, excluding the LCMP Type and LCMP Length fields of the sub-option.

3.1.1. Binding Re-registration Control Sub-Option

The Binding Re-registration Control Sub-Option is a mobility sub-option carried in the LMA Controlled MAG Parameters mobility option defined in Section 3.1. This sub-option carries re-registration related timer values. There MUST be no more than a single instance of this sub-option in LMA Controlled MAG Parameters option. The format of this sub-option is defined 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| LCMP Type = 1 |  LCMP Length  |  Re-registration-Start-Time   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Initial-Retransmission-Time  | Maximum-Retransmission-Time   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: Binding Re-registration Control Sub-Option

Re-registration-Start-Time

16-bit unsigned integer indicating the number of time units before the expiry of the PMIPv6 binding lifetime when the registration refresh process needs to be activated. One time unit is 4 seconds.

Initial-Retransmission-Time

16-bit unsigned integer indicating minimum delay in seconds before the first PBU retransmission of the exponential back-off process.

Maximum-Retransmission-Time

16-bit unsigned integer indicating maximum delay in seconds before the last PBU retransmission message of the exponential back-off process.

3.1.2. Heartbeat Control Sub-Option

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | LCMP Type = 2 |  LCMP Length  |        HB-Interval            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |    HB-Retransmission-Delay    |    HB-Max-Retransmissions     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: Heartbeat Control Sub-Option

HB-Interval

16-bit unsigned integer indicating heartbeat interval, i.e. time delay in seconds after a successful heartbeat exchange (request followed by response) when the next heartbeat exchange can be triggered.

HB-Retransmission-Delay

16-bit unsigned integer indicating minimum time delay in seconds before a heartbeat message is retransmitted.

HB-Max-Retransmissions

16-bit unsigned integer indicating maximum number of heartbeat retransmissions.

4. Protocol Configuration Variables

4.1. Local Mobility Anchor - Configuration Variables

The local mobility anchor MUST allow the following variables to be configured by the system management. The configured values for these protocol variables MUST survive server reboots and service restarts.

EnableLCMPSubOptReregControl

This flag indicates the operational state of the Binding Re-registration Control sub-option support. The default value for this flag is set to (0), indicating that support for the Binding Re-registration Control sub-option is disabled.

When this flag on the local mobility anchor is set to a value of (1), the local mobility anchor SHOULD include this sub-option in the Proxy Binding Acknowledge messages that it sends to the mobile access gateway; otherwise, it MUST NOT include the sub-option. There can be situations where the local mobility anchor is unable to obtain the Binding Re-registration Control information and may not be able to construct this sub-option.

EnableLCMPSubOptHeartbeatControl

This flag indicates the operational state of the Heartbeat Control sub-option support. The default value for this flag is set to (0), indicating that support for the Heartbeat Control sub-option is disabled.

When this flag on the local mobility anchor is set to a value of (1), the local mobility anchor SHOULD include this sub-option in the Proxy Binding Acknowledge messages that it sends to the mobile access gateway; otherwise, it MUST NOT include the sub-option. There can be situations where the local mobility anchor is unable to obtain the Heartbeat Control information and may not be able to construct this sub-option.

The following variables MAY be defined at various granularity such as per binding, per peering MAG, per cluster of MAGs or any other custom grouping. Regardless of the granularity of this configuration, the local mobility anchor should be able to determine the value of these variables on an individual binding basis by way of configuration hierarchy.

LCMPReregistrationStartTime

This variable is used to set the minimum time interval in number of seconds before the expiry of the PMIPv6 binding lifetime when the registration refresh process SHOULD be activated. The default value is 10 units, where each unit is 4 seconds.

LCMPInitialRetransmissionTime

This variable is used to set the minimum delay in seconds before the first PBU retransmission of the exponential back-off process. This variable is same as INITIAL_BINDACK_TIMEOUT mentioned in Section 6.9.4 of [RFC5213]. The default value is 1 second.

LCMPMaximumRetransmissionTime

This variable is used to set the maximum delay in seconds before the last PBU retransmission message of the exponential back-off process. This variable is same as MAX_BINDACK_TIMEOUT mentioned in Section 6.9.4 of [RFC5213]. The default value is 32 seconds.

LCMPHeartbeatInterval

This variable is used to set the time delay in seconds after a successful heartbeat exchange (request followed by response) when the next heartbeat exchange can be triggered. The default value is 60 seconds. It SHOULD NOT be set to less than 30 seconds or more than 3600 seconds. The value of this variable MAY be derived from the variable HEARTBEAT_INTERVAL defined in Section 5 of [RFC5847] if defined on the local mobility anchor.

LCMPHeartbeatRetransmissionDelay

This variable is used to set the minimum time delay in seconds before a heartbeat message is retransmitted. The value of this variable SHOULD be less than LCMP_HEARTBEAT_INTERVAL. The default value is 5 seconds.

LCMPHeartbeatMaxRetransmissions

This variable is used to set the maximum number of heartbeat retransmissions. The default value for this variable is 3. The value of this variable MAY be derived from the variable MISSING_HEARTBEATS_ALLOWED defined in Section 5 of [RFC5847] if defined on the local mobility anchor.

5. Protocol Considerations

The following considerations apply to the local mobility anchor and the mobile access gateway.

The conceptual Binding Cache Entry data structure maintained by the local mobility anchor, described in Section 5.1 of [RFC5213] and the conceptual Binding Update List entry data structure maintained by the mobile access gateway, described in Section 6.1 of [RFC5213], MUST be extended to store the LMA Controlled MAG Parameters option related information elements associated with the current session. Specifically the following parameters MUST be defined:

- LCMPReregistrationStartTime

- LCMPInitialRetransmissionTime

- LCMPMaximumRetransmissionTime

- LCMPHeartbeatInterval

- LCMPHeartbeatRetransmissionDelay

- LCMPHeartbeatMaxRetransmissions

5.1. Local Mobility Anchor Considerations

- On receiving a Proxy Binding Update message [RFC5213] from a mobile access gateway, the local mobility anchor should check if EnableLCMPSubOptReregControl is set to (1). If yes, and if all of LCMPReregistrationStartTime, LCMPInitialRetransmissionTime and LCMPMaximumRetransmissionTime are set to NON_ZERO values, then in SHOULD include Binding Re-registration Control Sub-Option in the LMA Controlled MAG Parameters mobility option which is in turn included in the Proxy Binding Acknowledge message.

- If EnableLCMPSubOptReregControl is set to (1) and if any of LCMPReregistrationStartTime, LCMPInitialRetransmissionTime and LCMPMaximumRetransmissionTime is set to ZERO value, then the local mobility anchor should report a configuration error.

- The local mobility anchor should also check if EnableLCMPSubOptHeartbeatControl is set to (1). If yes, and if all of LCMPHeartbeatInterval, LCMPHeartbeatRetransmissionDelay and LCMPHeartbeatMaxRetransmissions are set to NON_ZERO values, then in SHOULD include Heartbeat Control Sub-Option in the LMA Controlled MAG Parameters mobility option which is in turn included in the Proxy Binding Acknowledge message.

- If EnableLCMPSubOptHeartbeatControl is set to (1) and if any of LCMPHeartbeatInterval, LCMPHeartbeatRetransmissionDelay and LCMPHeartbeatMaxRetransmissions is set to ZERO value, then the local mobility anchor should report a configuration error.

5.2. Mobile Access Gateway Considerations

- On Receiving Proxy Binding Acknowledge message [RFC5213] from the local mobility anchor with LMA Controlled MAG Parameters mobility option, the mobile access gateway MUST overwrite the binding re-registration related timer parameters with the parameters received in Binding Re-registration Control Sub-Option, if present in the LMA Controlled MAG Parameters mobility option. Similarly, the mobile access gateway MUST overwrite the heartbeat related timer parameters with the parameters received in Heartbeat Control Sub-Option, if present in the LMA Controlled MAG Parameters mobility option.

- If any of the parameters in the Binding Re-registration Control Sub-Option is ZERO, then the sub-option MUST be ignored and an error message SHOULD be logged.

- If any of the parameters in the Heartbeat Control Sub-Option except HB-Retransmission-Delay is ZERO, then the sub-option MUST be ignored and error message SHOULD be logged.

6. IANA Considerations

This document requires the following IANA actions.

- Action 1: This specification defines a new mobility header option, the LMA Controlled MAG Parameters. This mobility option is described in Section 3. The type value (IANA-1) for this option needs to be assigned from the same numbering space as allocated for the other mobility options, as defined in [RFC6275].

- Action 2: This specification defines a new mobility sub-option format, the LMA Controlled MAG Parameters sub-option. The format of this mobility sub-option is described in Section 3.1. This sub-option can be carried in the LMA Controlled MAG Parameters option. The type value for this sub-option needs to be managed by IANA, under the registry "LMA Controlled MAG Parameters Sub-Option Type Values". This specification reserves the following type values. Approval of new LMA Controlled MAG Parameters sub-option type values are to be made through IANA Expert Review.

0	Reserved
1	Binding Re-registration Control Sub-Option
2	Heartbeat Control Sub-Option

7. Security Considerations

The LMA Controlled MAG Parameters option defined in this specification is for use in Proxy Binding Acknowledgement message. This option is carried like any other mobility header option as specified in [RFC6275] and does not require any special security considerations.

8. References

8.1. 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, <http://www.rfc-editor.org/info/rfc2119>.

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

[RFC5847] Devarapalli, V., Ed., Koodli, R., Ed., Lim, H., Kant, N., Krishnan, S., and J. Laganier, "Heartbeat Mechanism for Proxy Mobile IPv6", RFC 5847, DOI 10.17487/RFC5847, June 2010, <http://www.rfc-editor.org/info/rfc5847>.

[RFC7563] Pazhyannur, R., Speicher, S., Gundavelli, S., Korhonen, J., and J. Kaippallimalil, "Extensions to the Proxy Mobile IPv6 (PMIPv6) Access Network Identifier Option", RFC 7563, DOI 10.17487/RFC7563, June 2015, <http://www.rfc-editor.org/info/rfc7563>.

8.2. Informative References

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, DOI 10.17487/RFC2460, December 1998, <http://www.rfc-editor.org/info/rfc2460>.

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

Authors' Addresses

Dhananjay Patki
Cisco
Cessna Business Park SEZ, Kadubeesanahalli
Bangalore, Karnataka 560087
India

Email: dhpatki@cisco.com

Sri Gundavelli
Cisco
170 West Tasman Drive
San Jose, CA 95134
USA

Email: sgundave@cisco.com

Jong-Hyouk Lee
Sangmyung University
31, Sangmyeongdae-gil, Dongnam-gu
Cheonan 330-720
Republic of Korea

Email: jonghyouk@smu.ac.kr

Qiao Fu
China Mobile
Xuanwumenxi Ave. No.32
Beijing
China

Email: fuqiao1@outlook.com

Lyle T Bertz
Sprint
Kansas
USA

Email: Lyle.T.Bertz@sprint.com

DMM WG P. Seite
Internet-Draft Orange
Intended status: Standards Track A. Yegin
Expires: January 26, 2017 Samsung
S. Gundavelli
Cisco
July 25, 2016

MAG Multipath Binding Option
draft-ietf-dmm-mag-multihoming-02.txt

Abstract

The document [RFC4908] proposes to rely on multiple Care-of Addresses (CoAs) capabilities of Mobile IP [RFC6275] an Network Mobility (NEMO; [RFC3963]) to enable Multihoming technology for Small-Scale Fixed Networks. In the continuation of [RFC4908], this document specifies a multiple proxy Care-of Addresses (pCoAs) extension for Proxy Mobile IPv6 [RFC5213]. This extension allows a multihomed Mobile Access Gateway (MAG) to register more than one proxy care-of-address to the Local Mobility Anchor (LMA).

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Using several links, the multihoming technology can improve connectivity availability and quality of communications; the goals and benefits of multihoming are as follows:

- Redundancy/Fault-Recovery
- Load balancing
- Load sharing
- Preferences settings

According to [RFC4908], users of Small-Scale Networks can take benefit of multihoming using mobile IP [RFC6275] and Network Mobility (NEMO) [RFC3963] architecture in a mobile and fixed networking environment. This document is introducing the concept of multiple Care-of Addresses (CoAs) [RFC5648] that have been specified since then.

In the continuation of [RFC4908], a Proxy Mobile IPv6 [RFC5213] based multihomed achitecture could be defined. The motivation to update [RFC4908] with proxy Mobile IPv6 is to leverage on latest mobility working group achievments, namely:

- using GRE as mobile tuneling, possibly with its key extension [RFC5845] (a possible reason to use GRE is given on Section 3.2).
- using UDP encapsulation [RFC5844] in order to support NAT traversal in IPv4 networking environment.
- Prefix Delegation mechanism [RFC7148].
- Using the vendor specific mobility option [RFC5094], for example to allow the MAG and LMA to exchange information (e.g. WAN interface QoS metrics) allowing to make appropriate traffic steering decision.

Proxy Mobile IPv6 (PMIPv6) relies on two mobility entities: the mobile access gateway (MAG), which acts as the default gateway for the end-node and the local mobility anchor (LMA), which acts as the topological anchor point. Point-to-point links are established, using IP-in-IP tunnels, between MAG and LMA. Then, the MAG and LMA are distributing traffic over these tunnels. All PMIPv6 operations are performed on behalf of the end-node and its corespondent node, it thus makes PMIPv6 well adapted to multihomed architecture as considered in [RFC4908]. Taking the LTE and WLAN networking environments as an example, the PMIPv6 based multihomed architecture is depicted on Figure 1. Flow-1,2 and 3 are distributed either on Tunnel-1 (over LTE) or Tunnel-2 (over WLAN), while Flow-4 is spread on both Tunnel-1 and 2.

```
     Flow-1
      |
      |Flow-2              _----_
      | |         CoA-1  _(      )_   Tunnel-1
      | |    .---=======(   LTE    )========\ Flow-1
      | |    |           (_      _)          \Flow-4
      | |    |             '----'             \
      | | +=====+                              \  +=====+    _----_
      | '-|     |                               \ |     |  _(      )_
      '---| MAG |                                 | LMA |-( Internet )--
      .---|     |                                 |     |  (_      _)
      | .-|     |                               / |     |    '----'
      | | +=====+                              /  +=====+
      | |    |             _----_             /
      | |    |    CoA-2  _(      )_ Tunnel-2 /
      | |    .---=======(   WLAN  )========/ Flow-2
      | |                (_     _)           Flow-3
      | |                  '----'            Flow-4
      |Flow-3
      |
     Flow0-4
```

Figure 1: Multihomed MAG using Proxy Mobile IPv6

The current version of Proxy Mobile IPv6 does not allow a MAG to register more than one proxy Care-of-Adresse to the LMA. In other words, only one MAG/LMA link, i.e. IP-in-IP tunnel, can be used at the same time. This document overcomes this limitation by defining the multiple proxy Care-of Addresses (pCoAs) extension for Proxy Mobile IPv6.

2. Conventions and Terminology

2.1. Conventions

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].

2.2. Terminology

All mobility related terms used in this document are to be interpreted as defined in [RFC5213], [RFC5844] and [RFC7148]. Additionally, this document uses the following terms:

IP-in-IP

IP-within-IP encapsulation [RFC2473], [RFC4213]

3. Overview

3.1. Example Call Flow

Figure 2 is the callflow detailing multi-access support with PMIPv6. The MAG in this example scenario is equipped with both WLAN and LTE interfaces and is also configured with the multihoming functionality. The steps of the callflow are as follows:

Steps (1) and (2): the MAG attaches to both WLAN and LTE networks; the MAG obtains respectively two different proxy care-of-addresses (pCoA).

Step (3): The MAG sends, over the WLAN access, a Proxy Binding Update (PBU) message, with the new MAG Multipath Binding (MMB) and MAG Identifier (MAG-NAI) options to the LMA. A logical-NAI (MAG-NAI) with ALWAYS-ON configuration is enabled on the MAG. The mobility session that is created (i.e. create a Binding Cache Entry) on the LMA is for the logical-NAI. The LMA and allocates a Home Network Prefix (HNP), that shall be delegated to mobile nodes, to the MAG.

Step (4): the LMA sends back a Proxy Binding Acknowledgement (PBA) including the HNP allocated to the MAG.

Step (5): IP tunnel (IP-in-IP, GRE ...) is created over the WLAN access.

Steps (6) to (8): The MAG repeats steps (3) to (5) on the LTE access. The MAG includes the HNP, received on step (4) in the PBU. The LMA update its binding cache by creating a new mobility session for this MAG.

Steps (9) and (10): The IP hosts MN_1 and MN_2 are assigned IP addresses from the mobile network prefix delegated by the MAG.

```
   +=====+ +=====+     +=====+      +=====+      +=====+         +=====+
   | MN_1| | MN_2|     | MAG |      | WLAN|      | LTE |         | LMA |
   +=====+ +=====+     +=====+      +=====+      +=====+         +=====+
      |       |           |            |            |               |
      |       |           |            |            |               |
      |       |           | (1) ATTACH |            |               |
      |       |           | <--------> |            |               |
      |       |           | (2) ATTACH              |               |
      |       |           | <---------------------->|               |
      |       |           | (3) PBU (MAG-NAI, MMB)                  |
      |       |           | ------------------------*-------------->|
      |       |           |                                         |
      |       |           |                                   Accept PBU
      |       |           |                               (allocate HNP,
      |       |           |                                  create BCE)
      |       |           | (4) PBA (MAG-NAI, HNP)                  |
      |       |           | <-----------------------*---------------|
      |       |           | (5) TUNNEL INTERFACE CREATION over WLAN |
      |       |           |-============== TUNNEL ==*==============-|
      |       |           |                                         |
      |       |           | (6) PBU (MAG-NAI, HNP, MMB)             |
      |       |           | -----------*--------------------------->|
      |       |           |                                         |
      |       |           |                                   Accept PBU
      |       |           |                                 (update BCE)
      |       |           | (7) PBA (MAG-NAI, HNP)                  |
      |       |           | <----------*--------------------------- |
      |       |           | (8) TUNNEL INTERFACE CREATION over LTE  |
      |       |           |-===========*== TUNNEL =================-|
      |   (9) ATTACH      |                                         |
      | <---------------> |                                         |
      |       |(10) ATTACH|                                         |
      |       |<--------> |                                         |
```

Figure 2: Functional Separation of the Control and User Plane

3.2. Traffic distribution schemes

When receiving packets from the MN, the MAG distributes packets over tunnels that have been established. Traffic distribution can be managed either on a per-flow or on a per-packet basis:

- Per-flow traffic management: each IP flow (both upstream and downstream) is mapped to a given tunnel, corresponding to a given WAN interface. Flow binding extension [RFC6089] is used to exchange, and synchronize, IP flow management policies (i.e. rules associating traffic selectors [RFC6088] to a tunnel).

- Per-packet management: the LMA and the MAG distribute packets, belonging to a same IP flow, over more than one bindings (i.e. more than one WAN interface). When operating at the IP packet level, different packets distribution algorithms are possible. For example, the algorithm may give precedence to one given access: the MAG overflows traffic from the primary access, e.g. WLAN, to the second one, only when load on primary access reaches a given threshold. The distribution algorithm is left to implementer but whatever the algorithm is, packets distribution likely introduces packet latency and out-of-order delivery. LMA and MAG shall thus be able to make reordering before packets delivery. Sequence number can be can be used for that purpose, for example using GRE with sequence number option [RFC5845]. However, more detailed considerations on reordering and IP packet distribution scheme (e.g. definition of packets distribution algorithm) are out the scope of this document.

Because latency introduced by per-packet can cause injury to some application, per-flow and per-packet distribution schemes could be used in conjunction. For example, high throughput services (e.g. video streaming) may benefit from per-packet distribution scheme, while latency sensitive applications (e.g. VoIP) are not be spread over different WAN paths. IP flow mobility extensions, [RFC6089] and [RFC6088], can be used to provision the MAG with such flow policies.

4. Protocol Extensions

4.1. MAG Multipath-Binding Option

The MAG Multipath-Binding option is a new mobility header option defined for use with Proxy Binding Update and Proxy Binding Acknowledgement messages exchanged between the local mobility anchor and the mobile access gateway.

This mobility header option is used for requesting multipath support. It indicates that the mobile access gateway is requesting the local mobility anchor to register the current care-of address associated with the request as one of the many care-addresses through which the mobile access gateway can be reached. It is also for carrying the information related to the access network associated with the care-of address.

The MAG Multipath-Binding option has an alignment requirement of 8n+2. Its format is as shown in Figure 3:

```
    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      |    If-ATT     |    If-Label   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Binding-Id   |B|O|             RESERVED                      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: MAG Multipath Binding Option

Type

<IANA-1> To be assigned by IANA.

Length

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

Interface Access-Technology Type (If-ATT)

This 8-bit field identifies the Access-Technology type of the interface through which the mobile node is connected. The permitted values for this are from the Access Technology Type registry defined in [RFC5213].

Interface Label (If-Label)

This 8-bit field represents the interface label represented as an unsigned integer. The MAG identifies the label for each of the interfaces through which it registers a pCoA with the LMA. When using static traffic flow policies on the mobile node and the home agent, the label can be used for generating forwarding policies. For example, the operator may have policy which binds traffic for Application "X" needs to interface with Label "Y". When a registration through an interface matching Label "Y" gets activated, the home agent and the mobile node can dynamically generate a forwarding policy for forwarding traffic for Application "X" through mobile IP tunnel matching Label "Y". Both the home agent and the mobile node can route the Application-X traffic through that interface. The permitted values for If-Label are 1 through 255.

Binding-Identifier (BID)

This 8-bit field is used for carrying the binding identifier. It uniquely identifies a specific binding of the mobile node, to which this request can be associated. Each binding identifier is represented as an unsigned integer. The permitted values are 1 through 254. The BID value of 0 and 255 are reserved. The mobile access gateway assigns a unique value for each of its interfaces and includes them in the message.

Bulk Re-registration Flag (B)

This flag, if set to a value of (1), is to notify the local mobility anchor to consider this request as a request to update the binding lifetime of all the mobile node’s bindings, upon accepting this specific request. This flag MUST NOT be set to a value of (1), if the value of the Registration Overwrite Flag (O) is set to a value of (1).

Binding Overwrite (O)

This flag, if set to a value of (1), notifies the local mobility anchor that upon accepting this request, it should replace all of the mobile node’s existing bindings with this binding. This flag MUST NOT be set to a value of (1), if the value of the Bulk Re-registration Flag (B) is set to a value of (1). This flag MUST be set to a value of (0), in de-registration requests.

Reserved

This field is unused in this specification. The value MUST be set to zero (0) by the sender and MUST be ignored by the receiver.

4.2. MAG Identifier Option

The MAG Identifier option is a new mobility header option defined for use with Proxy Binding Update and Proxy Binding Acknowledgement messages exchanged between the local mobility anchor and the mobile access gateway. This mobility header option is used for conveying the MAG’s identity.

This option does not have any alignment requirements.

```
   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      |  Subtype      |  Reserved     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                           Identifier ...                      ~
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: MAG Identifier Option

Type

<IANA-2> To be assigned by IANA.

Length

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

Subtype

One byte unsigned integer used for identifying the type of the Identifier field. Accepted values for this field are the registered type values from the Mobile Node Identifier Option Subtypes registry.

Reserved

This field is unused in this specification. The value MUST be set to zero (0) by the sender and MUST be ignored by the receiver.

Identifier

A variable length identifier of type indicated in the Subtype field.

4.3. New Status Code for Proxy Binding Acknowledgement

This document defines the following new Status Code value for use in Proxy Binding Acknowledgement message.

CANNOT_SUPPORT_MULTIPATH_BINDING (Cannot Support Multipath Binding): <IANA-4>

5. IANA Considerations

This document requires the following IANA actions.

- Action-1: This specification defines a new mobility option, the MAG Multipath-Binding option. The format of this option is described in Section 4.1. The type value <IANA-1> for this mobility option needs to be allocated from the Mobility Options registry at <http://www.iana.org/assignments/mobility-parameters>. RFC Editor: Please replace <IANA-1> in Section 4.1 with the assigned value and update this section accordingly.

- Action-2: This specification defines a new mobility option, the MAG Identifier option. The format of this option is described in

Section 4.2. The type value <IANA-2> for this mobility option needs to be allocated from the Mobility Options registry at <http://www.iana.org/assignments/mobility-parameters>. RFC Editor: Please replace <IANA-2> in Section 4.2 with the assigned value and update this section accordingly.

- Action-3: This document defines a new status value, CANNOT_SUPPORT_MULTIPATH_BINDING (<IANA-3>) for use in Proxy Binding Acknowledgement message, as described in Section 4.3. This value is to be assigned from the "Status Codes" registry at <http://www.iana.org/assignments/mobility-parameters>. The allocated value has to be greater than 127. RFC Editor: Please replace <IANA-4> in Section 4.3 with the assigned value and update this section accordingly.

6. Security Considerations

This specification allows a mobile access gateway to establish multiple Proxy Mobile IPv6 tunnels with a local mobility anchor, by registering a care-of address for each of its connected access networks. This essentially allows the mobile node’s IP traffic to be routed through any of the tunnel paths and either based on a static or a dynamically negotiated flow policy. This new capability has no impact on the protocol security. Furthermore, this specification defines two new mobility header options, MAG Multipath-Binding option and the MAG Identifier option. These options are carried like any other mobility header option as specified in [RFC5213]. Therefore, it inherits security guidelines from [RFC5213]. Thus, this specification does not weaken the security of Proxy Mobile IPv6 Protocol, and does not introduce any new security vulnerabilities.

7. Acknowledgements

The authors of this draft would like to acknowledge the discussions and feedback on this topic from the members of the DMM working group.

8. References

8.1. 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, <http://www.rfc-editor.org/info/rfc2119>.

[RFC3963] Devarapalli, V., Wakikawa, R., Petrescu, A., and P. Thubert, "Network Mobility (NEMO) Basic Support Protocol", RFC 3963, DOI 10.17487/RFC3963, January 2005, <http://www.rfc-editor.org/info/rfc3963>.

[RFC5094] Devarapalli, V., Patel, A., and K. Leung, "Mobile IPv6 Vendor Specific Option", RFC 5094, DOI 10.17487/RFC5094, December 2007, <http://www.rfc-editor.org/info/rfc5094>.

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

[RFC5648] Wakikawa, R., Ed., Devarapalli, V., Tsirtsis, G., Ernst, T., and K. Nagami, "Multiple Care-of Addresses Registration", RFC 5648, DOI 10.17487/RFC5648, October 2009, <http://www.rfc-editor.org/info/rfc5648>.

[RFC5844] Wakikawa, R. and S. Gundavelli, "IPv4 Support for Proxy Mobile IPv6", RFC 5844, DOI 10.17487/RFC5844, May 2010, <http://www.rfc-editor.org/info/rfc5844>.

[RFC5845] Muhanna, A., Khalil, M., Gundavelli, S., and K. Leung, "Generic Routing Encapsulation (GRE) Key Option for Proxy Mobile IPv6", RFC 5845, DOI 10.17487/RFC5845, June 2010, <http://www.rfc-editor.org/info/rfc5845>.

[RFC6088] Tsirtsis, G., Giarreta, G., Soliman, H., and N. Montavont, "Traffic Selectors for Flow Bindings", RFC 6088, DOI 10.17487/RFC6088, January 2011, <http://www.rfc-editor.org/info/rfc6088>.

[RFC6089] Tsirtsis, G., Soliman, H., Montavont, N., Giaretta, G., and K. Kuladinithi, "Flow Bindings in Mobile IPv6 and Network Mobility (NEMO) Basic Support", RFC 6089, DOI 10.17487/RFC6089, January 2011, <http://www.rfc-editor.org/info/rfc6089>.

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

[RFC7148] Zhou, X., Korhonen, J., Williams, C., Gundavelli, S., and CJ. Bernardos, "Prefix Delegation Support for Proxy Mobile IPv6", RFC 7148, DOI 10.17487/RFC7148, March 2014, <http://www.rfc-editor.org/info/rfc7148>.

8.2. Informative References

[RFC2473] Conta, A. and S. Deering, "Generic Packet Tunneling in IPv6 Specification", RFC 2473, DOI 10.17487/RFC2473, December 1998, <http://www.rfc-editor.org/info/rfc2473>.

[RFC4213] Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", RFC 4213, DOI 10.17487/RFC4213, October 2005, <http://www.rfc-editor.org/info/rfc4213>.

[RFC4908] Nagami, K., Uda, S., Ogashiwa, N., Esaki, H., Wakikawa, R., and H. Ohnishi, "Multi-homing for small scale fixed network Using Mobile IP and NEMO", RFC 4908, DOI 10.17487/RFC4908, June 2007, <http://www.rfc-editor.org/info/rfc4908>.

Authors' Addresses

Pierrick Seite
Orange
4, rue du Clos Courtel, BP 91226
Cesson-Sevigne 35512
France

Email: pierrick.seite@orange.com

Alper Yegin
Samsung
Istanbul
Turkey

Email: alper.yegin@partner.samsung.com

Sri Gundavelli
Cisco
170 West Tasman Drive
San Jose, CA 95134
USA

Email: sgundave@cisco.com

DMM Working Group A. Yegin
Internet-Draft Actility
Intended status: Informational D. Moses
Expires: January 7, 2017 Intel
K. Kweon
J. Lee
J. Park
Samsung
July 6, 2016

On Demand Mobility Management
draft-ietf-dmm-ondemand-mobility-07

Abstract

Applications differ with respect to whether they need IP session continuity and/or IP address reachability. The network providing the same type of service to any mobile host and any application running on the host yields inefficiencies. This document describes a solution for taking the application needs into account in selectively providing IP session continuity and IP address reachability on a per-socket basis.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

In the context of Mobile IP [RFC5563][RFC6275][RFC5213][RFC5944], following two attributes are defined for the IP service provided to the mobile hosts:

IP session continuity: The ability to maintain an ongoing IP session by keeping the same local end-point IP address throughout the session despite the mobile host chaging its point of attachment within the IP network topology. The IP address of the host may change between two independent IP sessions, but that does not jeopardize the IP session continuity. IP session continuity is essential for mobile hosts to maintain ongoing flows without any interruption.

IP address reachability: The ability to maintain the same IP address for an extended period of time. The IP address stays the same across independent IP sessions, and even in the absence of any IP session. The IP address may be published in a long-term registry (e.g., DNS), and it is made available for serving incoming (e.g., TCP) connections. IP address reachability is essential for mobile hosts to use specific/published IP addresses.

Mobile IP is designed to provide both IP session continuity and IP address reachability to mobile hosts. Architectures utilizing these protocols (e.g., 3GPP, 3GPP2, WIMAX) ensure that any mobile host attached to the compliant networks can enjoy these benefits. Any application running on these mobile hosts is subjected to the same treatment with respect to the IP session continuity and IP address reachability.

It should be noted that in reality not every application may need those benefits. IP address reachability is required for applications running as servers (e.g., a web server running on the mobile host). But, a typical client application (e.g., web browser) does not necessarily require IP address reachability. Similarly, IP session continuity is not required for all types of applications either. Applications performing brief communication (e.g., DNS client) can survive without having IP session continuity support.

Achieving IP session continuity and IP address reachability by using Mobile IP incurs some cost. Mobile IP protocol forces the mobile host’s IP traffic to traverse a centrally-located router (Home Agent, HA), which incurs additional transmission latency and use of additional network resources, adds to the network CAPEX and OPEX, and decreases the reliability of the network due to the introduction of a single point of failure [I-D.ietf-dmm-requirements]. Therefore, IP session continuity and IP address reachability should be be provided only when needed.

Furthermore, when an application needs session continuity, it may be able to satisfy that need by using a solution above the IP layer, such as MPTCP [RFC6824], SIP mobility [RFC3261], or an application-layer mobility solution. Those higher-layer solutions are not subject to the same issues that arise with the use of Mobile IP since they can utilize the most direct data path between the end-points. But, if Mobile IP is being applied to the mobile host, those higher-layer protocols are rendered useless because their operation is inhibited by the Mobile IP. Since Mobile IP ensures the IP address of the mobile host remains fixed (despite the location and movement of the mobile host), the higher-layer protocols never detect the IP-layer change and never engage in mobility management.

This document proposes a solution for the applications running on the mobile host to indicate whether they need IP session continuity or IP address reachability. The network protocol stack on the mobile host, in conjunction with the network infrastructure, would provide the required type of IP service. It is for the benefit of both the users and the network operators not to engage an extra level of service unless it is absolutely necessary. So it is expected that applications and networks compliant with this specification would utilize this solution to use network resources more efficiently.

2. Notational Conventions

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 [RFC2119].

3. Solution

3.1. Types of IP Addresses

Three types of IP addresses are defined with respect to the mobility management.

- Fixed IP Address

A Fixed IP address is an address with a guarantee to be valid for a very long time, regardless of whether it is being used in any packet to/from the mobile host, or whether or not the mobile host is connected to the network, or whether it moves from one point-of-attachment to another (with a different subnet or IP prefix) while it is connected.

Fixed IP address are required by applications that need both IP session continuity and IP address reachability.

- Session-lasting IP Address

A session-lasting IP address is an address with a guarantee to be valid through-out the IP session(s) for which it was requested. It is guaranteed to be valid even after the mobile host had moved from one point-of-attachment to another (with a different subnet or IP prefix).

Session-lasting IP addresses are required by applications that need IP session continuity but do not need IP address reachability.

- Non-persistent IP Address

This type of IP address provides neither IP session continuity nor IP address reachability. The IP address is obtained from the serving IP gateway and it is not maintained across gateway changes. In other words, the IP address may be released and replaced by a new IP address when the IP gateway changes due to the movement of the mobile host.

Applications running as servers at a published IP address require a Fixed IP Address. Long-standing applications (e.g., an SSH session) may also require this type of address. Enterprise applications that connect to an enterprise network via virtual LAN require a Fixed IP Address.

Applications with short-lived transient IP sessions can use Session-lasting IP Addresses. For example: Web browsers.

Applications with very short IP sessions, such as DNS client and instant messengers, can utilize Non-persistent IP Addresses. Even though they could very well use a Fixed of Session-lasting IP Addresses, the transmission latency would be minimized when a Non-persistent IP Address is used.

The network creates the desired guarantee (Fixed, Session-lasting or Non-persistent) by either assigning an IP address (as part of a stateful IP address generation), or by assigning the address prefix (as part of a stateless address generation process).

3.2. Granularity of Selection

The IP address type selection is made on a per-socket granularity. Different parts of the same application may have different needs. For example, control-plane of an application may require a Fixed IP Address in order to stay reachable, whereas data-plane of the same application may be satisfied with a Session-lasting IP Address.

3.3. On Demand Nature

At any point in time, a mobile host may have a combination of IP addresses configured. Zero or more Non-persistent, zero or more Session-lasting, and zero or more Fixed IP addresses may be configured on the IP stack of the host. The combination may be as a result of the host policy, application demand, or a mix of the two.

When an application requires a specific type of IP address and such address is not already configured on the host, the IP stack shall attempt to configure one. For example, a host may not always have a Session-lasting IP address available. In case an application requests one, the IP stack shall make an attempt to configure one by issuing a request to the network. If the operation fails, the IP stack shall fail the associated socket request. If successful, a Session-lasting IP Address gets configured on the mobile host. If another socket requests a Session-lasting IP address at a later time, the same IP address may be served to that socket as well. When the last socket using the requested IP address is closed, the IP address may be released or kept for future applications that may be launched and require a Session-lasting IP address.

In some cases it might be preferable for the mobile host to request a new Session-lasting IP address for a new opening of an IP session (even though one was already assigned to the mobile host by the network and might be in use in a different, already active IP session). It is out of the scope of this specification to define criteria for selecting to use available addresses or choose to request new ones. It supports both alternatives (and any combination).

It is outside of the scope of this specification to define how the host requests a specific type of address (Fixed, Session-lasting or Non-persistent) and how the network indicates the type of address in its advertisement of addresses (or in its reply to an address request).

The following are matters of policy, which may be dictated by the host itself, the network operator, or the system architecture standard:

- The initial set of IP addresses configured on the host at the boot time.

- Permission to grant various types of IP addresses to a requesting application.

- Determination of a default address type when an application does not make any explicit indication, whether it already supports the required API or it is just a legacy application.

3.4. Conveying the Selection

The selection of the address type is conveyed from the applications to the IP stack in a way to influence the source address selection algorithm [RFC6724].

The current source address selection algorithm operates on the available set of IP addresses when selecting an address. According to the proposed solution, if the requested type IP address is not available at the time of the request, the IP stack shall make an attempt to configure one such IP address. The selected IP address shall be compliant with the requested IP address type, whether it is selected among available addresses or dynamically configured. In the absence of a matching type (because it is not available and not configurable on demand), the source address selection algorithm shall return an empty set.

A Socket API-based interface for enabling applications to influence the source address selection algorithm is described in [RFC5014]. That specification defines IPV6_ADDR_PREFERENCES option at the IPPROTO_IPV6 level. That option can be used with setsockopt() and getsockopt() calls to set and get address selection preferences.

Furthermore, that RFC also specifies two flags that relate to IP mobility management: IPV6_PREFER_SRC_HOME and IPV6_PREFER_SRC_COA. These flags are used for influencing the source address selection to prefer either a Home Address or a Care-of Address.

Unfortunately, these flags do not satisfy the aforementioned needs due to the following reasons, therefore new flags are proposed in this document:

- Current flags indicate a "preference" whereas there is a need for indicating "requirement". Source address selection algorithm does not have to produce an IP address compliant with the "preference" , but it has to produce an IP address compliant with the "requirement".

- Current flags influence the selection made among available IP addresses. The new flags force the IP stack to configure a compliant IP address if none is available at the time of the request.

- The Home vs. Care-of Address distinction is not sufficient to capture the three different types of IP addresses described in Section 2.1.

The following new flags are defined in this document and they shall be used with Socket API in compliance with the [RFC5014]:

IPV6_REQUIRE_FIXED_IP /* Require a Fixed IP address as source */

IPV6_REQUIRE_SESSION_LASTING_IP /* Require a Session-lasting IP address as source */

IPV6_REQUIRE_NON-PERSISTENT_IP /* Require a Non-persistent IP address as source */

Only one of these flags may be set on the same socket. If an application attempts to set more than one flag, the most recent setting will be the one in effect.

When any of these new flags is used, then the IPV6_PREFER_SRC_HOME and IPV6_PREFER_SRC_COA flags, if used, shall be ignored.

These new flags are used with setsockopt()/getsockopt(), getaddrinfo(), and inet6_is_srcaddr() functions [RFC5014]. Similar with the setsockopt()/getsockopt() calls, getaddrinfo() call shall also trigger configuration of the required type IP address, if one is not already available. When the new flags are used with getaddrinfo() and the triggered configuration fails, the getaddrinfo() call shall ignore that failure (i.e., not return an error code to indicate that failure). Only the setsockopt() shall return an error when configuration of the requested type IP address fails.

The following new error codes are also defined in the document and will be used in the Socket API in compliance with [RFC5014].

EAI_REQUIREDIPNOTSUPPORTED /* The network does not support the ability to request that specific IP address type */

EAI_REQUIREDIPFAILED /* The network could not assign that specific IP address type */

4. Backwards Compatibility Considerations

Backwards compatibility support is required by the following 3 types of entities:

- The Applications on the mobile host

- The IP stack in the mobile host

- The network infrastructure

4.1. Applications

Legacy applications that do not support the new flags will use the legacy API to the IP stack and will not enjoy On-Demand Mobility feature.

Applications using the new flags must be aware that they may be executed in environments that do not support On-Demand Mobility feature. Such environments may include legacy IP stack in the mobile host, legacy network infrastructure, or both. In either case, the API will return an error code and the invoking applications must respond with using legacy calls without On-Demand Mobility feature.

4.2. IP Stack in the Mobile Host

New IP stacks must continue to support all legacy operations. If an application does not use On-Demand Mobility feature, the IP stack must respond in a legacy manner.

If the network infrastructure supports On-Demand Mobility feature, the IP stack should follow the application request: If the application requests a specific address type, the stack should forward this request to the network. If the application does not request an address type, the IP stack must not request an address type and leave it to the network's default behavior to choose the type of the allocated IP address. If an IP address was already allocated to the host, the IP stack uses it and may not request a new one from the network.

4.3. Network Infrastructure

The network infrastructure may or may not support the On-Demand Mobility feature. How the IP stack on the host and the network infrastructure behave in case of a compatibility issue is outside the scope of this API specification.

5. Summary of New Definitions

The following list summarizes the new constants definitions discussed in this memo:

<netdb.h>	IPV6_REQUIRE_FIXED_IP
<netdb.h>	IPV6_REQUIRE_SESSION_LASTING_IP
<netdb.h>	IPV6_REQUIRE_NON_PERSISTENT_IP
<netdb.h>	EAI_REQUIREDIPNOTSUPPORTED
<netdb.h>	EAI_REQUIREDIPFAILED
<netinet/in.h>	IPV6_REQUIRE_FIXED_IP
<netinet/in.h>	IPV6_REQUIRE_SESSION_LASTING_IP
<netinet/in.h>	IPV6_REQUIRE_NON_PERSISTENT_IP
<netinet/in.h>	EAI_REQUIREDIPNOTSUPPORTED
<netinet/in.h>	EAI_REQUIREDIPFAILED

6. Security Considerations

The setting of certain IP address type on a given socket may be restricted to privileged applications. For example, a Fixed IP Address may be provided as a premium service and only certain applications may be allowed to use them. Setting and enforcement of such privileges are outside the scope of this document.

7. IANA Considerations

This document has no IANA considerations.

8. Acknowledgements

We would like to thank Alexandru Petrescu, John Kaippallimalil, Jouni Korhonen, Seil Jeon, and Sri Gundavelli for their valuable comments and suggestions on this work.

9. References

9.1. 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, <http://www.rfc-editor.org/info/rfc2119>.

[RFC5014] Nordmark, E., Chakrabarti, S., and J. Laganier, "IPv6 Socket API for Source Address Selection", RFC 5014, DOI 10.17487/RFC5014, September 2007, <http://www.rfc-editor.org/info/rfc5014>.

[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, <http://www.rfc-editor.org/info/rfc6724>.

9.2. Informative References

[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-17 (work in progress), June 2014.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <http://www.rfc-editor.org/info/rfc3261>.

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

[RFC5563] Leung, K., Dommety, G., Yegani, P., and K. Chowdhury, "WiMAX Forum / 3GPP2 Proxy Mobile IPv4", RFC 5563, DOI 10.17487/RFC5563, February 2010, <http://www.rfc-editor.org/info/rfc5563>.

[RFC5944] Perkins, C., Ed., "IP Mobility Support for IPv4, Revised", RFC 5944, DOI 10.17487/RFC5944, November 2010, <http://www.rfc-editor.org/info/rfc5944>.

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

[RFC6824] Ford, A., Raiciu, C., Handley, M., and O. Bonaventure, "TCP Extensions for Multipath Operation with Multiple Addresses", RFC 6824, DOI 10.17487/RFC6824, January 2013, <http://www.rfc-editor.org/info/rfc6824>.

Authors’ Addresses

Alper Yegin
Actility
Istanbul
Turkey

Email: alper.yegin@actility.com

Danny Moses
Intel Corporation
Petah Tikva
Israel

Email: danny.moses@intel.com

Kisuk Kweon
Samsung
Suwon
South Korea

Email: kisuk.kweon@samsung.com

Jinsung Lee
Samsung
Suwon
South Korea

Email: js81.lee@samsung.com

Jungshin Park
Samsung
Suwon
South Korea

Email: shin02.park@samsung.com

DMM Working Group   D. Moses
Internet-Draft   Intel
Intended status: Standards Track   A. Yegin
Expires: March 31, 2017   September 27, 2016

DHCPv6 Extension for On Demand Mobility exposure
draft-moses-dmm-dhcp-ondemand-mobility-04

Abstract

Applications differ with respect to whether or not they need IP session continuity and/or IP address reachability. Networks providing the same type of service to any mobile host and any application running on the host yields inefficiencies. This document describes extensions to the DHCPv6 protocol to enable mobile hosts to indicate the required mobility service type associated with a requested IP address, and networks to indicate the type of mobility service associated with the allocated IP address in return.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

[I-D.ietf-dmm-ondemand-mobility] defines different types of mobility-associated services provided by access networks to mobile hosts with regards to maintaining IPv6 address continuity after an event of the host moving to different locations with different points of attachments within the IP network topology. It further specifies means for applications to convey to the IP stack in the mobile host, their requirements regarding these services.

This document defines extensions to the DHCPv6 protocol ([RFC3315]) in the form of a new DHCP option that specifies the type of mobility services associated with an IPv6 address. The IP stack in a mobile host uses the DHCP client to communicate the type of mobility service it wishes to receive from the network. The DHCP server in the network uses this option to convey the type of service that is guaranteed with the assigned IPv6 address in return.

This new option also extends the ability of mobile routers to specify desired mobility service in a request for IPv6 proxies (as specified in [RFC3633]), and delegating routers to convey the type of mobility service that is committed with the allocated IPv6 proxies in return.

In a distributed mobility management environment, there are multiple Mobility Anchors (as specified in [TBD reference to the Distributed Mobility Management architecture RFC]). In some use-cases, mobile hosts may wish to indicate to the network, preference of the serving Mobility Anchor. This document specifies a new DHCPv6 option that is used by DHCPv6 clients to convey this preference.

2. Notational Conventions

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 [RFC2119].

3. IPv6 Continuity Service Option

The IPv6 Continuity Service option is used to specify the type of continuity service associated with a source IPv6 address or IPv6 prefix. The IPv6 Continuity Service option must be encapsulated in the IAaddr-options field of the IA Address option when associated with an IPv6 address, and in the IAprefix-options field of the IA_PD prefix option when associated with an IPv6 prefix.

The format of the IPv6 Continuity Service option is:

```
 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_IPv6_CONTINUITY_SERVICE|         option-length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| service-type  |
+-+-+-+-+-+-+-+-+
```

option-code   OPTION_IPv6_CONTINUITY_SERVICE (TBD)

option-len   1

service-type   one of the following values:

Non-Persistent -   a non-persistent IP address or prefix (1)

Session-Lasting -   a session-lasting IP address or prefix (2)

Fixed -   a fixed IP address or prefix (3)

Anytype -   Anyone of the above (0)

The definition of these service types is available in [I-D.ietf-dmm-ondemand-mobility].

All other values (4-255) are reserved for future usage and should not be used. If the OPTION_IPv6_CONTINUITY_SERVICE option is received and its service-type is equal to one of the reserved values, the option should be ignored.

This option can appear in one of two contexts: (1) As part of a request to assign a source IPv6 address of the specified mobility service type, and (2) As part of a request to assign an IPv6 prefix of the specified mobility service type.

3.1. Source IPv6 Address Type Specification

In this context, the IPv6 Continuity Service option is encapsulated in the IAaddr-options field of the IA Address option.

When in a message sent from a client to a server, the value of the IPv6 Continuity Service option indicates the type of continuity service required for the IPv6 address requested by the client.

When in a message sent from a server to a client, the value of the IPv6 Continuity Service option indicates the type of IP continuity service committed by the network for the associated IPv6 address. The value 'AnyType' can only appear in the message sent from the client to the server to indicate that the client has no specific preference. However, it cannot appear in a message sent from the server.

Once an IPv6 address type was requested and provided, any subsequent messages involving this address (lease renewal - for example) must include the IPv6 Continuity Service option with the same service type that was assigned by the server during the initial allocation.

If a server received a request to assign an IPv6 address with a specified IPv6 Continuity service, but cannot fulfill the request, it must reply with the NoAddrsAvail status (refer to section 22.13 - Status Code Option in [RFC3315].

A server that does not support this option will discard it as well as the IA Address option that had this option encapsulated in one of its IAaddr-options field.

If a client does not receive the requested address, it must resend the request without the desired IPv6 Continuity Service option since it is not supported by the server. In that case, the host of the client cannot assume any IP continuity service behaviour for that address.

A server must not include the IPv6 Continuity Service option in the IAaddr-options field of an IA Address option, if not specifically requested previously by the client to which it is sending a message.

If a client receives an IA Address option from a server with the IPv6 Continuity Service option in the IAaddr-options field, without initially requesting a specific service using this option, it must discard the received IPv6 address.

If the mobile host has no preference regarding the type of continuity service it uses the 'AnyType' value as the specified type of continuity service. The Server will allocate an IPv6 address with some continuity service and must specify the type in IPv6 Continuity Service option encapsulated in the IAaddr-options field of the IA Address option. The method for selecting the type of continuity service is outside the scope of this specification.

3.2. IPv6 Prefix Type Specification

In this context, the IPv6 Continuity Service option is encapsulated in the IAprefix-options field of the IA_PD prefix option.

When in a message sent from a client to a server, the value of the IPv6 Continuity Service option indicates the type of continuity service required for the IPv6 prefix requested by the client.

When in a message sent from a server to a client, the value of the IPv6 Continuity Service option indicates the type of continuity service committed by the network for the associated IPv6 prefix. The value 'AnyType' can only appear in the message sent from the client to the server to indicate that the client has no specific preference. However, it cannot appear in a message sent from the server.

Once an IPv6 prefix type was requested and provided, any subsequent messages involving this prefix (lease renewal - for example) must include the IPv6 Continuity Service option with the same service type that was assigned by the server during the initial allocation.

If a server received a request to assign an IPv6 prefix with a specified IPv6 Continuity service, but cannot fulfill the request, it must reply with the NoAddrsAvail status.

A server that does not support this option will discard it as well as the IA_PD Prefix option that had this option encapsulated in one of its IAprefix-options field.

If a client does not receive the requested prefix, it must resend the request without the desired IPv6 Continuity Service option since it is not supported by the server. In that case, the mobile router of the client cannot assume any IP continuity service behaviour for that prefix.

A server must not include the IPv6 Continuity Service option in the IAprefix-options field of an IA_PD Prefix option, if not specifically requested previously by the client to which it is sending a message.

If a client receives an IA_PD Prefix option from a server with the IPv6 Continuity Service option in the IAprefix-options field, without initially requesting a specific service using this option, it must discard the received IPv6 prefix.

If the mobile router has no preference regarding the type of continuity service it uses the ’AnyType’ value as the specified type of continuity service. The Server will allocate an IPv6 prefix with some continuity service and must specify the type in IPv6 Continuity Service option encapsulated in the IAprefix-options field of the IA_PD Prefix option. The method for selecting the type of continuity service is outside the scope of this specification.

4. Anchor Preference Option

In a distributed mobility management environment that deploys multiple Mobility Anchors, each Mobility Anchor may have a set of IPv6 prefixes that is being used when assigning Session-lasting or Fixed source IPv6 addresses to hosts.

The selection of the Mobility Anchor that will serve a mobile host is performed by the network at various events like, the event of initial attachment of a mobile host to a network.

The Anchor Preference option enables a host to express its desire to receive a source IPv6 address with a specific IPv6 prefix. This is useful when the mobile host wishes to indicate to the network which Mobility Anchor should be used for anchoring its traffic and ensuring service continuity in the event of handoff between LANs with different IPv6 prefixes.

The network MAY respect this request but is not required to do so.

The format of the Anchor Preference option is:

```
    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_ANCHOR_PREFERENCE      |         option-length         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                      preferred-lifetime                       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | prefix6len    |          ipv6-prefix                          |
   +-+-+-+-+-+-+-+-+       (variable length)                       |
   .                                                               .
   |-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                  IAanchor_preference-options                  |
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

option-code   OPTION_ANCHOR_PREFERENCE (TBD)

option-len   5 + length of ipv6-prefix field + length of anchor_preference-options field

preferred-lifetime The preferred lifetime of the IPv6 address whose prefix is requested, expressed in units of seconds

prefix-length The length in bits of the ipv6-prefix. Typically allowed values are 0 to 128.

IPv6 prefix   This is a variable length field that specifies the desired ipv6 prefix. The length is (prefix6len + 7) / 8. This field is padded with 0 bits up to the nearest octet boundary when prefix6len is not divisible by 8.

IAanchor_preference-option Options associated with this request

This option is used by the client in a request for a new IPv6 source address. The server replies with an IPv6 address that may or may not have the desired prefix.

An IPv6 prefix is requested only when the mobile host wishes to be anchored by a specific mobility anchor. The client must also indicate the type of mobility service it requires using the IPv6 Continuity Service option encapsulated in the IAanchor_preference-options field of the IA Address option.

When requesting an IPv6 prefix, only the ’Session-Lasting’ and ’Fixed’ types are legal.

The server must assign the IPv6 address of the requested type to the client, even if it does not fulfill the request for the specified prefix.

If a server received a request to use a specific IPv6 prefix and an IPv6 address type, but cannot assign an IPv6 address with that specified IPv6 Continuity it must reply with the NoAddrsAvail status.

A server that does not support this option will discard it.

If a client does not receive any address, it must assume that the the option is not supported by the server and use the IA Address option in subsequent requests.

5. Security Considerations

There are no specific security considerations for this option.

6. IANA Considerations

TBD

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, <http://www.rfc-editor.org/info/rfc2119>.

7.2. Informative References

[I-D.ietf-dmm-ondemand-mobility]
Yegin, A., Moses, D., Kweon, K., Lee, J., and J. Park, "On Demand Mobility Management", draft-ietf-dmm-ondemand-mobility-07 (work in progress), July 2016.

[RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July 2003, <http://www.rfc-editor.org/info/rfc3315>.

[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, <http://www.rfc-editor.org/info/rfc3633>.

Authors' Addresses

Danny Moses
Intel
Petah Tikva
Israel

Email: danny.moses@intel.com

Alper Yegin
Istanbul
Turkey

Email: alper.yegin@yegin.org

DMM Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 4, 2017

S. Jeon
Sungkyunkwan University
S. Figueiredo
Altran Research
Y. Kim
Soongsil University
J. Kaippallimalil
Huawei

October 31, 2016

Use Cases and API Extension for Source IP Address Selection
draft-sijeon-dmm-use-cases-api-source-05.txt

Abstract

This draft specifies and analyzes the expected cases regarding the selection of a proper source IP address and address type by an application in a distributed mobility management (DMM) network. It also proposes a new Socket API to address further selection issues with three source IP address types defined in the on-demand mobility API draft.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Applications to select source IP address type in a mobile node (MN) need to consider IP session continuity and/or IP address reachability. [I-D.ietf-dmm-ondemand-mobility], defines three types of source IP addresses based on mobility management capabilities: fixed IP address, session-lasting IP address, and non-persistent IP address. Based on the address type requested by the application, the MN configures a proper source IP address. However, the source IP address type itself in a socket request may not be enough to convey all the requirements of an application. For example, more than one IP address of the same type requested may be available. It may be that as a result of mobility the MN can potentially obtain new IP prefixes from different serving networks belonging to different subnets. This draft categorizes and analyzes use cases that an MN is likely to encounter. In addition, this draft proposes an extension that allows the application to express its preferences when more than one source address of a type is present.

2. Use Cases and Analysis

This section outlines use cases where an application on the MN tries to obtain a source IP address.

2.1. Application has no specific IP address type requirement or address preference

Applications such as text-based web browsing or information service, e.g. weather and stock information, as well as legacy applications belong to this category. Many applications use short-lived Internet connections with no requirements for session continuity or IP address reachability. Assigning a non-persistent IP address can be thus considered as default for MNs. However, it is subject to address assignment policy of a network operator. The suggested flag, IPV6_REQUIRE_NON-PERSISTENT_IP, defined in [I-D.ietf-dmm-ondemand-mobility] can used for expressing its preference to the IP stack.

2.2. Application has specific IP address type requirement and address preference

This category is for an application requiring IP session continuity with different granularity of IP address reachability. This case may be further divided in three sub-cases with regard to IP address type availability and/or address selection.

2.2.1. Case 1: there is no configured IP address based on a requested type in the IP stack, but there is a further selection preference by the application

Once an IP address is requested by an application regardless of any source IP address type defined in [I-D.ietf-dmm-ondemand-mobility], the network stack will configure an IP address after obtaining an IP prefix based on the requested source IP address type from the current serving gateway.

2.2.2. Case 2: there are one or more IP addresses configured with a requested type in the IP stack, and no selection preference by the application

This is the same as Case 1 described above, except the existence of more than one configured IP addresses belonging to the requested IP address type in the IP stack, e.g. due to different address assignment policy by an operator.

When a non-persistent IP address is requested, if an application requests a non-persistent IP address to the IP stack, the IP address is obtained from the serving IP gateway as the previous one is not maintained across gateway changes.

When a session-lasting IP address is requested, an expected sequence can be described as follows;

1. The MN has one or more session-lasting IP addresses configured in the IP stack.

2. If an application requests a session-lasting IP address to the IP stack, it will try to use an existing session-lasting IP address as it is already configured in the IP stack. If there are multiple available session-lasting IP addresses, the default address selection rules will be applied [RFC6724], e.g. with scope preference, longest prefix matching, and/or so on. The best-matched IP address among them will be selected and assigned to the application.

3. Subsequently, the MN moves to another serving network, and the previous (mobile) sessions are still in use. A new application requests a session-lasting IP address with flag, IPV6_REQUIRE_SESSION_LASTING_IP to the IP stack. The selection of the session-lasting IP address follows the same procedure as described in Step 2.

When a fixed IP address is requested, it will follow the same procedure with session-lasting IP address request as described.

2.2.3. Case 3: there are one or more IP addresses with a requested type configured in the IP stack, but there is a further selection preference by the application

Assume that there are one or multiple applications with session-lasting IP address running. A newly initiated application might get one of the session-lasting IP addresses being used, not initiating a protocol procedure, i.e. DHCP or SLAAC for a new session-lasting IP address to the network. On the contrary, the IP stack might try to get a new session-lasting IP address from the current serving gateway by default. Acquiring a new session-lasting IP address may take some time (due to the exchange with the network) while using an existing one is instantaneous. On the other hand, using the existing one might yield less optimal routing. For example, the use of the IP address with an existing one configured might provide a suboptimal routing path as a result of a handover. This situation might not be preferred by newly initiated applications because the application incurs the costs of IP mobility even though the MN has not moved from the current serving network. Eventually, the new session is served by a remote IP mobility anchor with mobility management functions, though the MN has not moved yet.

If the application is allowed to further define its preference for an optimally routed, this situation can be avoided. See Section 3 for the proposed flag.

2.3. Gaps in the consistency with the default address selection

The need of an indication mechanism can be sought in the consistency with the former IETF standards. For example, in [RFC6724] where default behavior for IPv6 is specified, without a proper indication mechanism, following conflicts are expected to happen. In Rule 6 in [RFC6724], it is said that the matching label between source address of an IPv6 host and destination address is preferred among the combinations between other source addresses and destination address, where the label is a numeric value representing policies that prefer a particular source address prefix for use with a destination address prefix in [RFC6724]. In Rule 8 in [RFC6724], it is said that the longest matching prefix between source address of an IPv6 host and destination address is preferred among the combinations between other source addresses and destination address. Following Rules 6 and 8 may result in the selection of a source IP address with which packets that are sub-optimally routed.

3. Indications for expressing address preference requirement

When an application prefers a new IP address of the requested IP address type, additional indication flags should be delivered through the socket API interface.

To obtain an address that supports dynamic mobility using session-lasting IP address, a new address preference flag needs to be defined. The flag should be simple and useful while aligned with the three types of IP addresses. The objective of the hereby presented address preference flag is letting the IP stack check whether it has an available IP address assigned from the current serving network when the flag is received by an initiated application. If not, it will trigger the IP stack to get a new IP address from the current serving network. We call it "ON_NET" property.

If the application requests an IP address with ON_NET flag set in the socket request, the IP address returned by the stack should conform to the address preference requirement. This should be the case even though other session-lasting IP addresses, not belonging to the current serving network are available. If there are multiple session-lasting IP addresses matched with ON_NET property, the default source address selection rules will be applied.

IPV6_XX_SRC_ON_NET

/* Require (or Prefer) an IP address based on a requested IP address type as source, assigned from the current serving network, whatever it has been assigned or should be assigned */

This flag aims to express the preference to check an IP address, being used by an application, previously assigned from the current serving network and to use it or to get an IP address from the current serving network, as well as enabling differentiated per-flow anchoring where an obtained session-lasting IP address might be used for all initiated session-lasting IP applications. The use of the flag can be combined together with the three types of IP address defined in [I-D.ietf-dmm-ondemand-mobility].

In [I-D.mccann-dmm-prefixcost], it proposes that the Router Advertisement signaling messages communicate the cost of maintaining a given prefix at the MN’s current point of attachment. The objective is to make a dynamic and optimal decision of address assignment and release, i.e. when to release old addresses and assign new ones. The proposed ON_NET property may present a way to deliver a prefix decision for an application, specifically from a routing distance point of view, to the IP stack.

4. IANA Considerations

This document makes no request of IANA.

5. Security Considerations

T.B.D.

6. Acknowledgements

We would like to thank Danny Moses, Marco Liebsch, Brian Haberman, Sri Gundavelli, Alexandru Petrescu for their valueable comments and suggestions on this work.

7. References

7.1. Normative References

[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, <http://www.rfc-editor.org/info/rfc6724>.

7.2. Informative References

[I-D.ietf-dmm-ondemand-mobility]
Yegin, A., Moses, D., Kweon, K., Lee, J., and J. Park, "On Demand Mobility Management", draft-ietf-dmm-ondemand-mobility-07 (work in progress), July 2016.

[I-D.mccann-dmm-prefixcost]
McCann, P. and J. Kaippallimalil, "Communicating Prefix Cost to Mobile Nodes", draft-mccann-dmm-prefixcost-03 (work in progress), April 2016.

Authors’ Addresses

Seil Jeon
Sungkyunkwan University
2066 Seobu-ro, Jangan-gu
Suwon, Gyeonggi-do
Korea

Email: seiljeon@skku.edu

Sergio Figueiredo
Altran Research
2, Rue Paul Dautier
Velizy-Villacoublay 78140
France

Email: sergio.figueiredo@altran.com

Younghan Kim
Soongsil University
369, Sangdo-ro, Dongjak-gu
Seoul 156-743
Korea

Email: younghak@ssu.ac.kr

John Kaippallimalil
Huawei
5340 Legacy Dr., Suite 175
Plano, TX 75024
U.S

Email: john.kaippallimalil@huawei.com