Basic and Advanced features of IPv6 Over C2C NET

Manabu Tsukada, K. Yacine, Thierry Ernst

To cite this version:

Manabu Tsukada, K. Yacine, Thierry Ernst. Basic and Advanced features of IPv6 Over C2C NET. [Research Report] 2009. inria-00565574

HAL Id: inria-00565574
https://hal.inria.fr/inria-00565574

Submitted on 14 Feb 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Basic and Advanced features of IPv6 Over C2C NET

Manabu Tsukada, Yacine Khaled and Thierry Ernst

July 31, 2009

Abstract

The GeoNet project will significantly contribute to vehicle communication by implementing a reference specification of a geographic addressing and routing protocol with support for IPv6 to be used to deliver safety messages between cars but also between cars and the roadside infrastructure within a designated destination area. Geographic addressing and routing is a networking mechanism distributing the information to nodes within a designated destination area. A novel routing protocol (C2C NET) is in charge of information dissemination over multiple hops until every vehicle has received this information within the destination area. This document mentions about basic and advanced features of IPv6 over C2C NET. First, we discover the missing features in current specification of C2C NET and shows some solutions. Second, specification of IPv6 over C2C NET are described and implementation example is investigated in Linux system. Third, we propose advance features such as route optimization, multihoming and simultaneous utilization of NEMO and C2C NET V2V mode.

Contents

Part I

Overview of IPv6 over C2C-NET link

1 Assumptions

- The IPv6 address of the OBU is formed by a network prefix and a C2C NET ID.
- A given IPv6 prefix uniquely identifies a given C2C NET link (i.e. a dedicated geographic area around an AR)

1.1 C2C Architecture

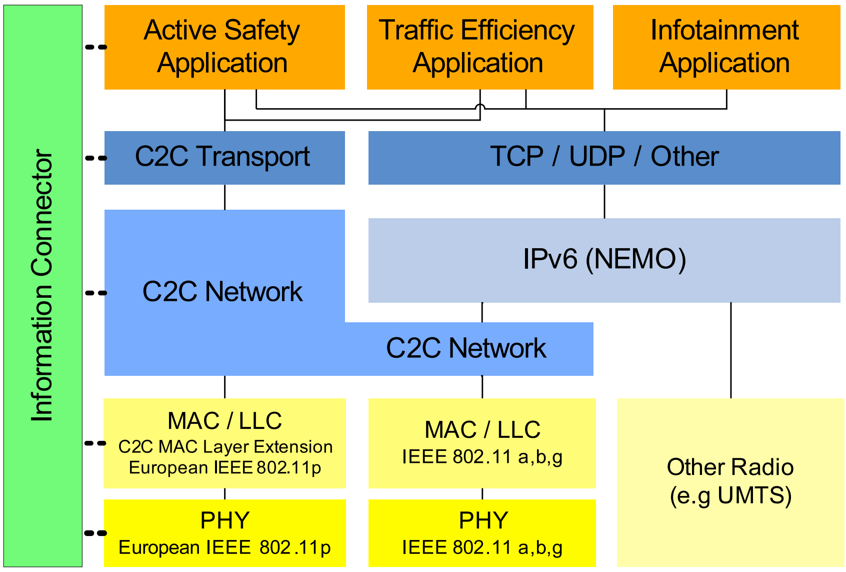


Figure 1: C2C Architecture

1.2 C2C NET Identifier

The Interface Identifier [1] for an Ethernet interface is based on the EUI-64 identifier derived from the interface's built- in 48-bit IEEE 802 address. The C2C NET ID is the interface Identifier with 64-bit address uniquely assigned in each C2C Interface.

1.3 C2C Header

1.3.1 Geo-unicast packet

Figure 2 shows the Geo-unicast packet format. Protocol type 2, Protocol subtype 0, hop limit 255 and packet length are included in C2C Common header. The location information for each entry shall include at least all the position information in the common C2C header, it is called **position vector**, and its has the following data fields:

- C2C NET ID
- Timestamp
- Position in latitude, longitude and altitude along with their accuracy
- Speed and header along with their accuracy

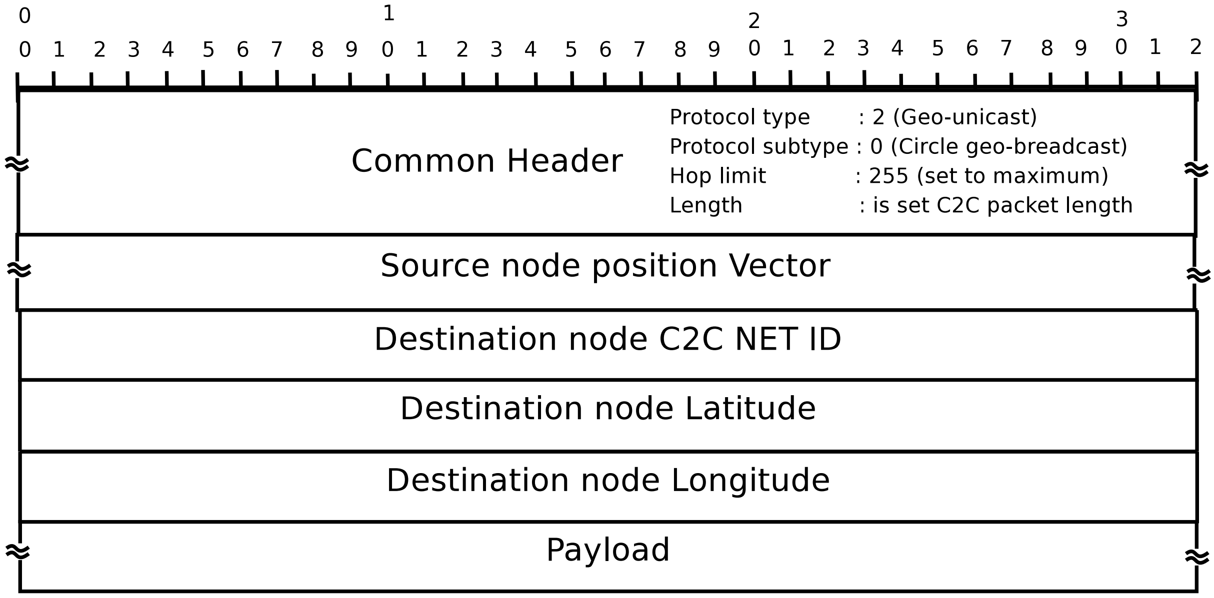


Figure 2: Geo-unicast packet

1.3.2 Geo-broadcast packet

Figure 3 shows the Geo-broadcast packet format. Protocol type 4, Protocol subtype 0 (circle geo-broadcast), hop limit 255 and packet length are included in C2C Common header. The location information for each entry shall include at least all the position information in the common C2C header, it is called **position vector**, and its has the following data fields:

- C2C NET ID
- Timestamp
- Position in latitude, longitude and altitude along with their accuracy
- Speed and header along with their accuracy

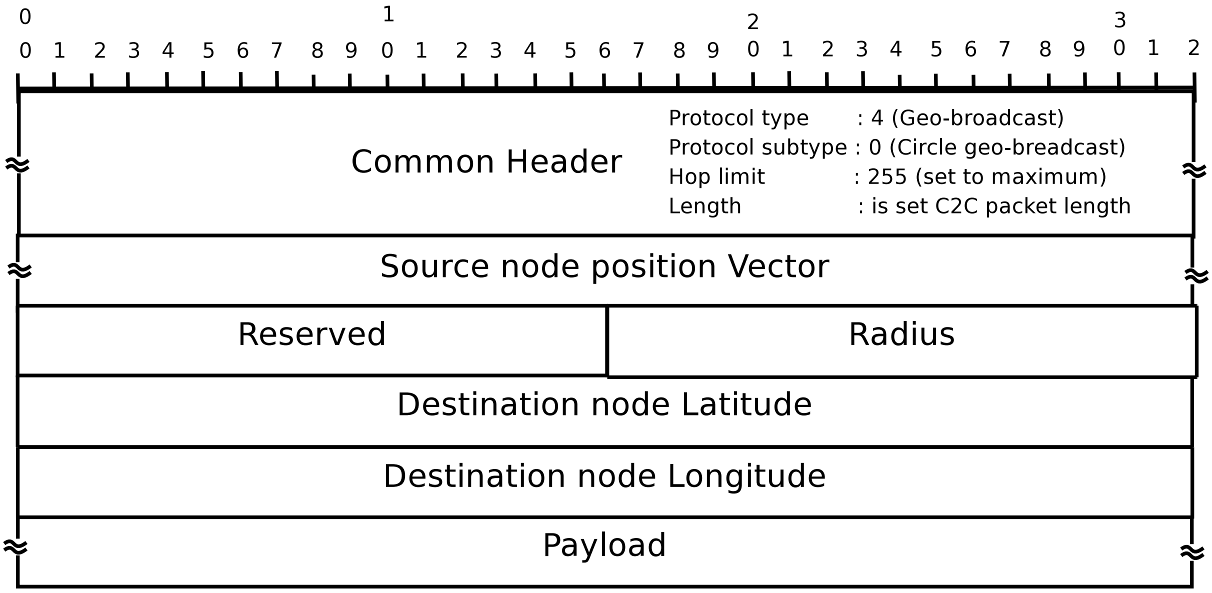


Figure 3: Geo-broadcast packet

2 V2V Scenario

The packet delivery in a V2V Geo-Unicast Scenario is illustrated in Figure 4. The IPv6 packet sent from AU1 is encapsulated with C2C NET header at OBU1. The intermediate OBUs (OBU2 and OBU3 in Figure 4) handle the C2C NET packet at the C2C layer. The C2C NET packet is delivered to C2C destination (OBU4) and the C2C header is decapsulated at OBU4. The end nodes (AU1 and AU2) send IPv6 packets without any change from IPv6 standard and OBUs treat C2C NET header to enable geographic routing. The destination IPv6 address of the packet is the IPv6 global address of AU2.

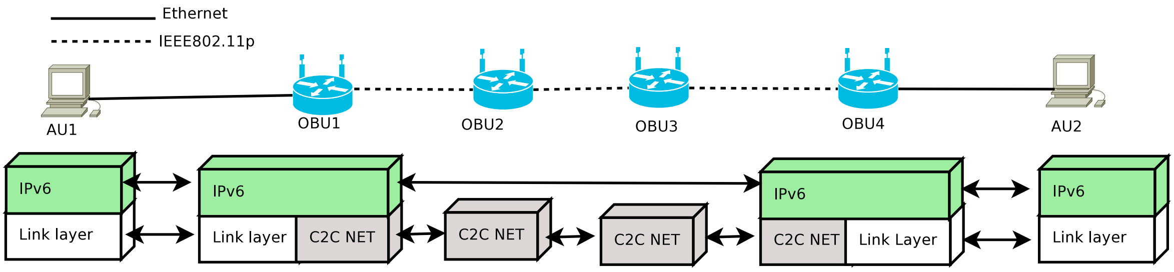


Figure 4: V2V Geo-Unicast Scenario

3 V2I Scenario

The V2I Geo-Unicast Scenario packet delivery is illustrated in Figure 5. The packets sent from AU1 are encapsulated with IP header as described in [2] in OBU1 and at the same time encapsulated with C2C NET header. The packets are delivered to RSU by geographic routing and the RSU decapsulates the C2C header in the packets and forwards it to the Internet. The packets are delivered to HA. The HA delivers the received packets (after removing the outer IPv6 header of the tunnel) to its final destination (the CN).

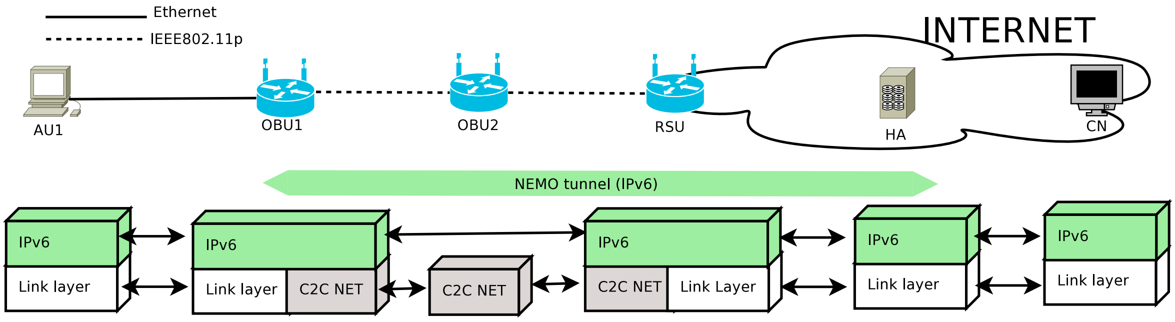


Figure 5: V2I Geo-Unicast Scenario

Part II

C2C NET Link: Problems and Approachs

4 Introduction

To deliver the packet to the right destination vehicle, the OBUs should know the route to the destination global address. Rest of the Part I is organized as follows: Section 5 describes the missing features of in-vehicle network discovery in C2C NET. Then three related works are investigated as a solution space analysis in section 6. Section 7 described five solutions to solve the problem and compare all of them. Section 17 concludes the Part I of the document.

5 Problem Statement for In-vehicle Network Discovery

Figure 6 illustrates the issue for IP next hop discovery. The C2C NET layer provides the routing functionality between OBUs. AUs have their IPv6 global addresses configured from the IPv6 prefix announced in their respective in-vehicle networks (Mobile Network Prefix, MNP). Now, AU1 starts communicating with AU3, thus the IPv6 source address is MNP1::AU1 and the IPv6 destination address is MNP3::AU3. When the packet arrives to OBU1, OBU1 should be able to determine that AU3 is reachable over the C2C-NET link. In other words, OBU1 should be able to determine that the next IPv6 hop to IPv6 destination address (MNP3::AU2) is OBU3. As a result from this determination, OBU1 should add the C2C NET header, which source C2C NET ID is C2CID1, and destination C2C NET ID is C2CID3. The problem is how to perform this determination given that MNPs reachable over the C2C-NET link are unknown and that OBUs have no information to match an IPv6 destination address to a given C2C-NET ID.

6 Solution space analysis

To solve the issue described in section 5, several approaches can be considered. We describe related work that may give us hint for the solution such as IP resolution on Ethernet, Host and Network Association (HNA) on OLSR and Router advertisement in Neighbor Discovery Protocol (NDP).

6.1 Routing and IP resolution over Ethernet and over C2C NET

Let's briefly describe about routing and IP resolution in Ethernet with figure 7. Router1 and Router2 are connected with one another over an Ethernet link. When Router1 receives a packet sent to Prefix2::/64, it finds out that this prefix is reachable through Router2 (Prefix3::R2) in the IP routing table. The route entry for that prefix was actually added to the routing table beforehand by **static** or **dynamic** means. Popular dynamic routing protocols are, for example, OSPF [3] or RIPng [4].

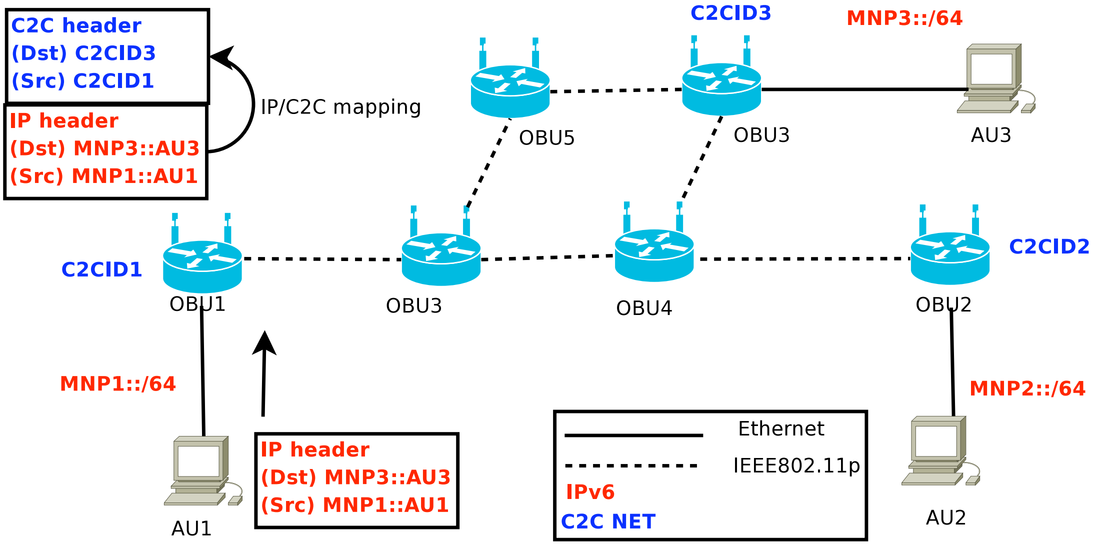


Figure 6: in-vehicle network discovery

Then, Router1 resolves MAC address of Router2 (MAC2 in figure7) from IP address of Router2 (Prefix3::R2) in the neighbor cache. When there is no entry in the neighbor cache for the resolution, Router1 sends Neighbor Solicitation (NS) on the link and Router2 replies with a Neighbor Advertisement (NA) message as defined in Neighbor Discovery Protocol (NDP)[5].

On the C2C NET link there are intermediate OBUs between source and destination. However we can keep the same routing table and neighbor cache architecture as the Ethernet case by assuming that all source and destination OBUs have their egress interface configured with an IPv6 address based on the C2C NET prefix (Prefix3::/64) . In this case, the IP routes should be known by OBUs and obtained by static or dynamic means, while IP resolution is not necessary because the IPv6 address of OBUs' egress interface always contain the C2C NET ID information. In figure 7, IP address of OBU2 (Prefix3::C2CID2) contains OBU's C2C NET ID (C2CID2). How to exchange routes between OBUs are still in question.

Note that one more alternative can be considered by introducing a brand-new mapping table which directly maps between MNP and C2C ID. However this needs new forwarding mechanism to lookup the new mapping table and requires big kernel implementation, which is not favorable.

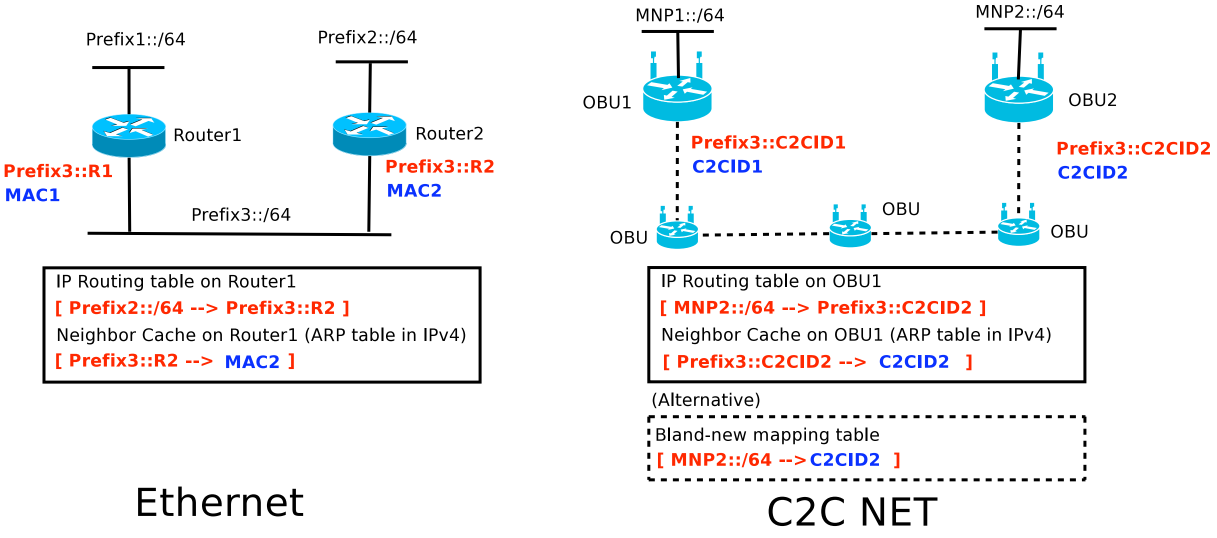


Figure 7: Routing and IP resolution in Ethernet and C2C NET

6.2 Host and Network Association (HNA) on OLSR

OLSR [6] provides Host and Network Association (HNA) [1] to find network behind OLSR nodes which connect to non-OLSR interface. HNA message is typically used for discovering in-vehicle network in the case of vehicular network because in-vehicle nodes may not have OLSR functionality. HNA message gives us a hint for the approach to the issue described in section 5.

Example of OLSR network is illustrated in Figure 8. OLSR routing is based on IPv4 or IPv6 addressing. Here is the example of IPv6 network, thus the alphabets from A to G marked in the figure 8 are IPv6 addresses of OLSR interfaces. By sending HNA message periodically, OLSR nodes can discover the network prefix behind of OLSR nodes. i.e. MNP1::/64 and MNP2::/64.

On the other hand, C2C NET doesn't have HNA message at this moment. Thus each OBU doesn't know the network behind OBUs while C2C NET IDs are known to one

---

[1] **What about HNA messages?** — RFC3626: OLSR, section 12 (page 51)

"A node MAY be equipped with multiple interfaces, some of which do not participate in the OLSR MANET. These non OLSR interfaces may be point to point connections to other singular hosts or may connect to separate networks.

In order to provide connectivity from the OLSR MANET interface(s) to these non OLSR interface(s), a node SHOULD be able to inject external route information to the OLSR MANET.

Injecting routing information from the OLSR MANET to non OLSR interfaces is outside the scope of this specification. It should be clear, however, that the routing information for the OLSR MANET can be extracted from the topology table (see section 4.4) or directly from the routing table of OLSR, and SHOULD be injected onto the non OLSR interfaces following whatever mechanism (routing protocol, static configuration etc.) is provided on these interfaces.

An example of such a situation could be where a node is equipped with a fixed network (e.g., an Ethernet) connecting to a larger network as well as a wireless network interface running OLSR.

Notice that this is a different case from that of "multiple interfaces", where all the interfaces are participating in the MANET through running the OLSR protocol.

In order to provide this capability of injecting external routing information into an OLSR MANET, a node with such non-MANET interfaces periodically issues a Host and Network Association (HNA) message, containing sufficient information for the recipients to construct an appropriate routing table."

another. C2C NET also needs HNA-like messaging to discover the network behind OBUs. Once in-vehicle network prefix are exchanged by HNA like message, the information should be stored in routing table as illustrated in Figure 8. The next hop information must be added to routing table in the form of IPv6 address, because routing table cannot understand C2C NET ID. The IPv6 address is generated with the network prefix of egress interface and C2CID. The prefix is marked as 'Prefix1::/64' in figure 8. 'Prefix1::/64' can be global prefix or link-local prefix (fe80::/64).

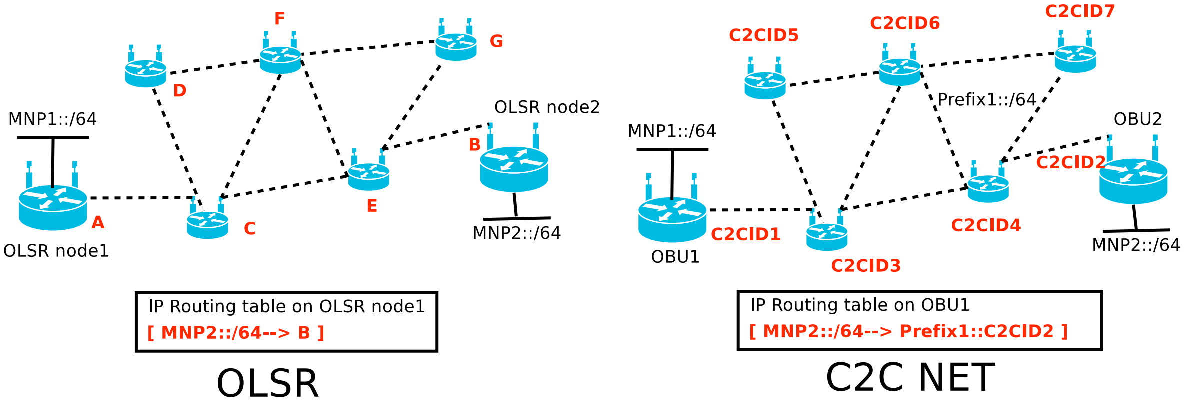


Figure 8: Host and Network Association (HNA) on OLSR

6.3 Router advertisement in Neighbor Discovery Protocol (NDP)

Router advertisement (RA) defined in NDP [5] is an example to find the network behind of C2C NET node, because OBUs understand that Internet is connected via source of RA (RSU) by the message. The process is illustrated in Figure 9. The RSU advertises RAs periodically into the C2C NET. The RAs are brought by geographic routing of C2C NET in certain area. When OBU1 receives the RA, it configures default route to link-local address of RSU (fe80::C2CID3). Notice that network information of ::0/0 is not contained in the RA because RA message it is not designed to notify other connected networks of the RA sender but on-link prefix (Prefix1::/64). Prefix1::/64 is usually a global routable prefix.

Extension of RA can be considered so that OBUs perform same process as RSU. In other words, OBU1 adds the second entry (MNP2::/64 –> fe80::C2CID2) to routing table by receiving the extended RA from OBU2 in figure 9. In order to notify the in-vehicle network prefix to the other OBUs, new RA option should be defined for C2C NET (RFC 4861 allows it[2].). Originally NDP is designed "*to discover each other's presence, to determine each other's link-layer addresses, to find routers, and to maintain reachability information about the paths to active neighbors* [3]". However the proposed new option enables to discover the network behind the neighbor. The extension uses RA message for

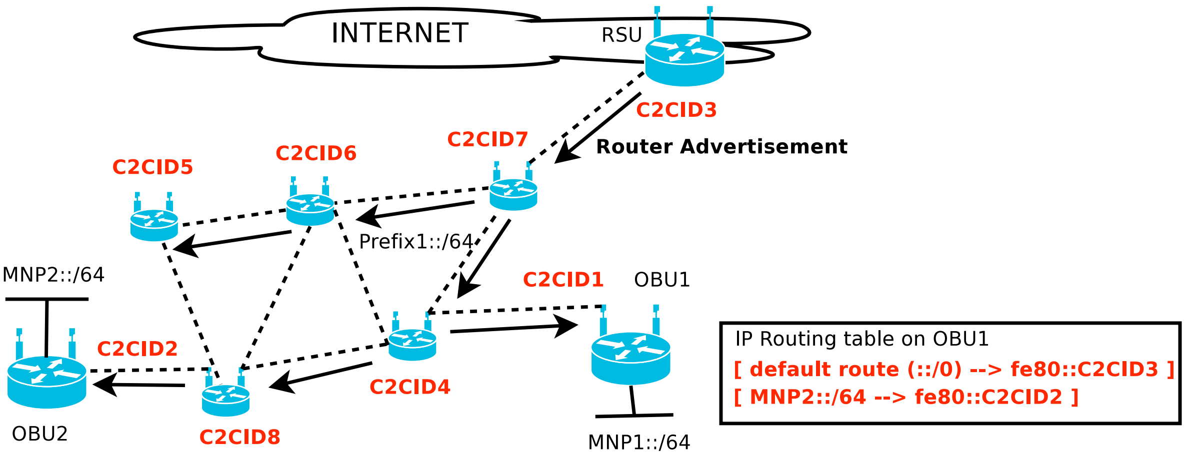
---

[2] **RFC 4861 section 4.2. Router Advertisement Message Format**
Future versions of this protocol may define new option types. Receivers MUST silently ignore any options they do not recognize and continue processing the message.

[3] Abstract of RFC4861

discovering the network behind of the RA sender which is different usage from the NDP's original design (on-link network advertisement).

The solution is debatable.



IP Routing table on OBU1
[default route (::/0) --> fe80::C2CID3]
[MNP2::/64 --> fe80::C2CID2]

Figure 9: Router advertisement in Neighbor Discovery Protocol (NDP)

7 Approaches Analysis

The solution analysis in section 6 gave us hints to approach the problem to discover the network behind OBUs. We've considered five approaches listed follows from the hints.

7.1 Approach description

7.1.1 Static route configuration

Static route configuration is the simplest approach to discover in-vehicle networks as investigated in section 6.1. Such static configuration is commonly used in Ethernet type of network which doesn't change its topology frequently. However, it is not suitable for C2C NET due to its mobility nature and due to scalability concerns. In order to use a static configuration in C2C NET, we can statically configure OBU1 like as follows in Figure 7 on Linux system.

```
# route -A inet6 add MNP2::/64 gw Prefix3::C2CID2 dev eth2
```

7.1.2 Route exchange by means of a dynamic routing protocol

Dynamic routing protocols such as OSPF or RIPng are popular solutions all over the Internet. We can employ it over C2C NET as investigated in section 6.1, because C2C NET is since as an IPv6 link from the IP layer view point.

7.1.3 HNA-like extension of C2C NET

We borrowed the idea of approach from HNA message of OLSR. In-vehicle network behind of OBUs are bound to C2C NET ID with the message as investigated in section 6.2. The message should be newly implemented in C2C NET. The propagation of the message can be optimized by geographic routing protocol of C2C NET.

7.1.4 RA extension for in-vehicle network discovery

The approach is investigated in section 6.3. As described before, the solution is debatable because the extension uses RA message to discover the network behind the RA sender which is a different usage from the NDP's original design (on-link network advertisement). In addition, the message is periodically broadcasted in C2C NET in certain area.

7.1.5 DNS-like in-vehicle network discovery

The approach is not investigated, however matching between in-vehicle network and C2C NET ID is not realized only by message exchange in C2C NET, but by message exchange between server and client like DNS. Matching time delay could be considerable.

7.2 Approach comparison

The five approaches are compared in table 1. First, we compare the additional works needed from a 'Difficulty' view point.

Static and Dynamic routing need less work for us than the others because we can reuse the implementations. The other approaches require some more works.

Second, scalability is the important issue to deploy vehicular networks. A static means doesn't scale at all, on the other hand all other approaches are able to scale, at least in some extend. The messaging between OBUs (approach 2,3 and 4) can be limited within a certain area with geographic routing capability of C2C NET, so message storm in C2C NET can be avoided. DNS-like approach needs a lot of queries to discover the in-vehicle networks, but DNS servers can be distributed. The scalability is proofed in the Internet.

Third, signaling overhead is compared. Static route configuration doesn't require any signaling between OBUs. The periodic advertisement approaches (approach 2,3 and 4) make signaling depending on the message frequency that is pre-configured. HNA-like extension may be able to optimize the messaging by using geographic information in C2C NET (managing frequency from geographic information, etc), while dynamic routing protocol and RA extension could not be optimized this way because these approaches stand on upper layer than C2C NET layer. DNS-like approach needs a query for one in-vehicle network discovery.

Forth, delay to find the in-vehicle network behind OBUs is compared. Static configuration has no delay. Periodic message exchange approaches (2,3 and 4) have delay depending on message frequency and RTT between OBUs. DNS-like approach has delay depending on RTT between client and server.

Table 1: Approach comparison

Approach	Difficulty	Scalability	Signaling overhead	Delay
(1) Static route configuration	easy	no	no overhead	no delay
(2) Dynamic routing protocol	easy	yes	periodic (a lot)	depend on frequentcy and RTT
(3) HNA like extension	hard	yes	optimized	depend on frequentcy and RTT
(4) RA extension	hard	yes	periodic (a lot)	depend on frequentcy and RTT
(5) DNS like discovery	hard	yes	once for each OBU	Server-client RTT

8 Conclusion

The straightforward approach is to implement HNA-like messaging at the C2C NET layer because it is the fundamental function for IPv6 over C2C NET. If it is not possible for some reason, the second best approach is a dynamic routing protocol at the IP layer, because it is easier than a RA extension approach and has the same benefits. In addition, a static route configuration is too ad-hoc solution and would never work in real life C2C NET environment. DNS-like discovery needs to be investigated.

Honestly, a static means is favorable to avoid considerable work for demonstration purposes, however it should not be written in the specification. If we use static route configuration in the demonstration, it should be secret for the audience. :)

Part III

Specification of IPv6 over C2C NET

9 SAP between IPv6 and C2C NET

The purpose of this SAP is to transmit the packet up from the C2C-NET layer to IP layer and down from IP layer to C2C-NET layer.

9.1 Type of destinations

As illustrated in Table 2, four types of destinations are considered. Each type of destination is assured by both IPv6 mechanism and C2C NET routing protocol. For instance, to reach a specific vehicle both IPv6 unicast and C2C NET geo-unicast are required.

Table 2: Types of destinations

Destination	IPv6 layer	C2C NET layer
A node in a specific vehicle	unicast	geo-unicast
Nodes in vehicles in area	multicast	geo-broadcast
Nodes in vehicles x hops away	mutlicast	topo-broadcast
A node in a certain vehicle in area	anycast	geo-anycast

9.2 From IP layer to C2C NET layer

According to Table 2, only one function, nammed GeoIPv6, is needed to transmit the packet from IP layer to C2C NET layer. In this function three parameters could be considered: scope, destination and payload.

- scope: the exact number of scopes depends on the assumption that IP layer knows which C2C NET routing protocol is needed, especially between geo-broadcast and topo-broadcast. Thus, in the case when IP layer doesn't know C2C NET routing protocol, only 3 kinds of scope are needed (unicast, anycast and multicast), otherwise, 4 scopes are needed (unicast, anycast, geo-broadcast and topo-broadcast) (see Table 3).
- destination: In unicast, IP layer provides, to C2C NET layer, IP next hop as destination address. Geo-Routing module determines C2C-NET ID from IP next hop. On the other hand, for IP multicast and anycast, areaID is provided to C2C NET layer. areaID could be either position or distance and shall be embedded in the multicast address. (See also Figure. 10)
- payload: contains IP packet.

Table 3: Parameters of GeoIPv6 function

destination\Parameters	scope	destination	payload
A nodes in a specific vehicle	unicast	IP next hop	IP packet
Nodes in vehicles in area	geo-broadcast	Area ID, Radius	IP packet
Nodes in vehicles x hops away	topo-broadcast	Hop limit	IP packet
A nodes in certain vehicle in area	anycast	Area ID, Radius	IP packet

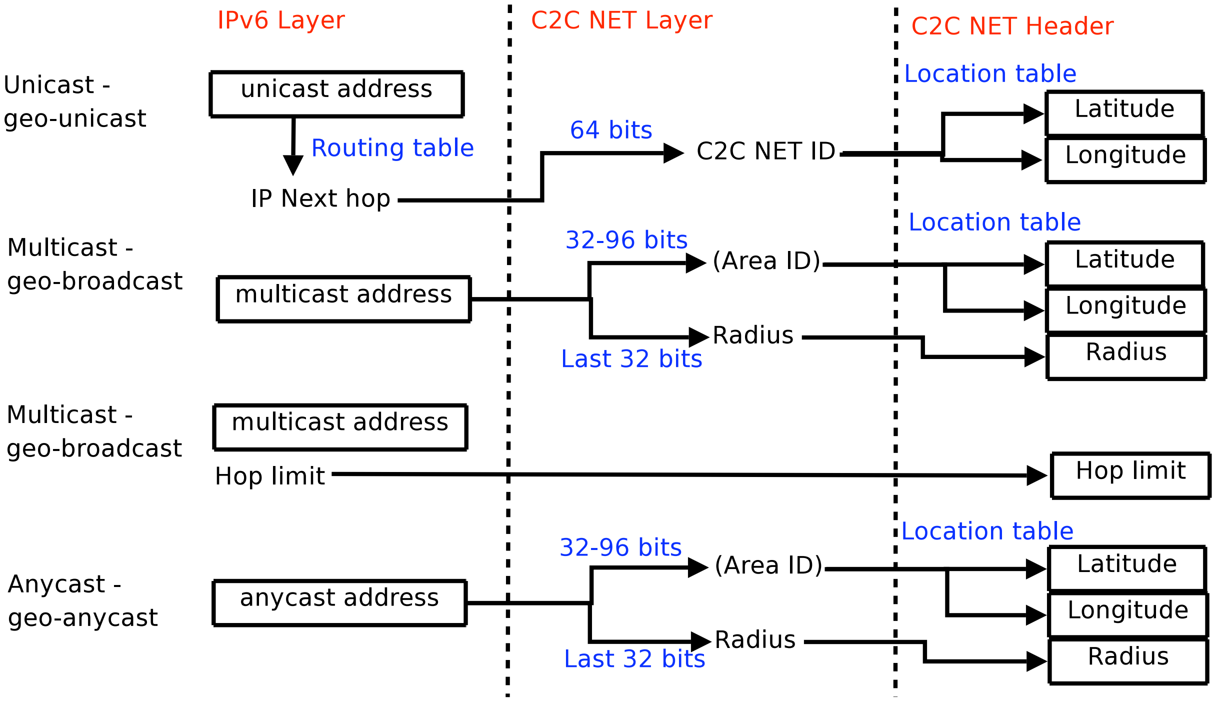


Figure 10: Service Access Point between IPv6 and C2C NET

10 IPv6 Unicast Using Geo-Unicast

In order to ensure IPv6 unicast packet transmission through C2C NET layer, IP layer should provide some information to C2C NET layer such as destination address and IP packet. IP unicast address transmitted to the C2C-NET is the IP next hop, not final destination. In addition to full IP packet, IP next hop should be provided to the C2C-NET Geo-Routing module, since IP next hop is considered as destination in C2C NET viewpoint.

Only one function is needed to provide all the information from IP layer to C2C NET layer. For instance, this function could be nammed GeoIPv6: GeoIPv6 (scope:unicast; dest:IPnexthop; payload:IP packet). The parameters of this function are described in section 9.

As illustrated in figure11, the unicast over C2C NET has two scenarios: unicast in certain C2C NET and unicast via the Internet . In the scenario with unicast in C2C NET, IP layer transmits, to C2C NET layer, the address of OBU2 as IP next hop in order to reach the destination AU2. However, in unicast via Internet scenario IP layer passes, to C2C NET layer, RSU1 address as IP next hop.

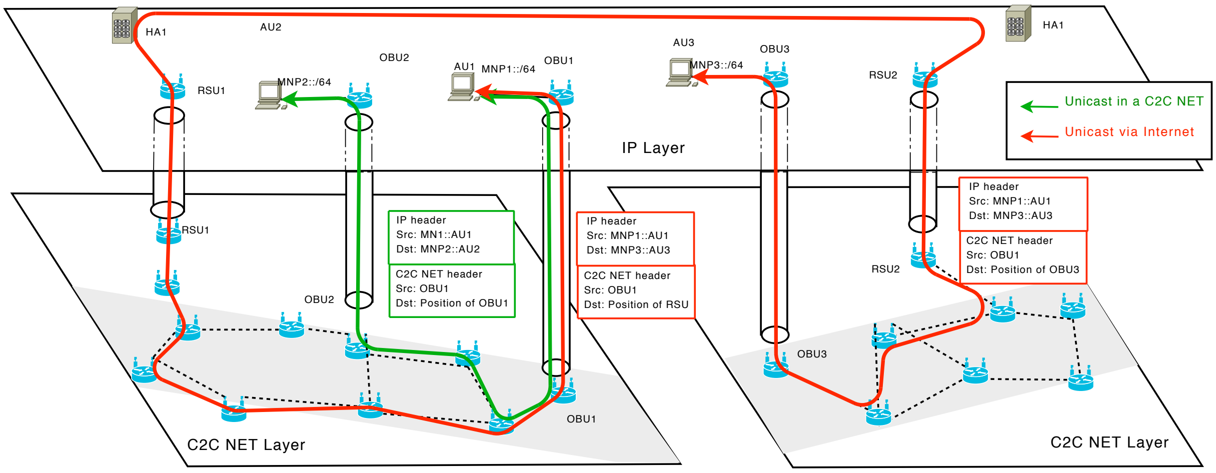


Figure 11: Unicast Over C2C NET

11 IPv6 Multicast Using Geo-Broadcast

11.1 Scenarios of Geo-Broadcast

Scenarios of Geo-Broadcast are divided to two scenarios by the destination of the multicast packets (Figure. 12). **Around Geo-broadcast** packet is delivered to the area around the source OBU while **Area Geo-broadcast** packet is delivered to the specific geographic area. Former only requires the radius to be specified and later requires all set of (latitude, longitude and radius) to delivered the packet.

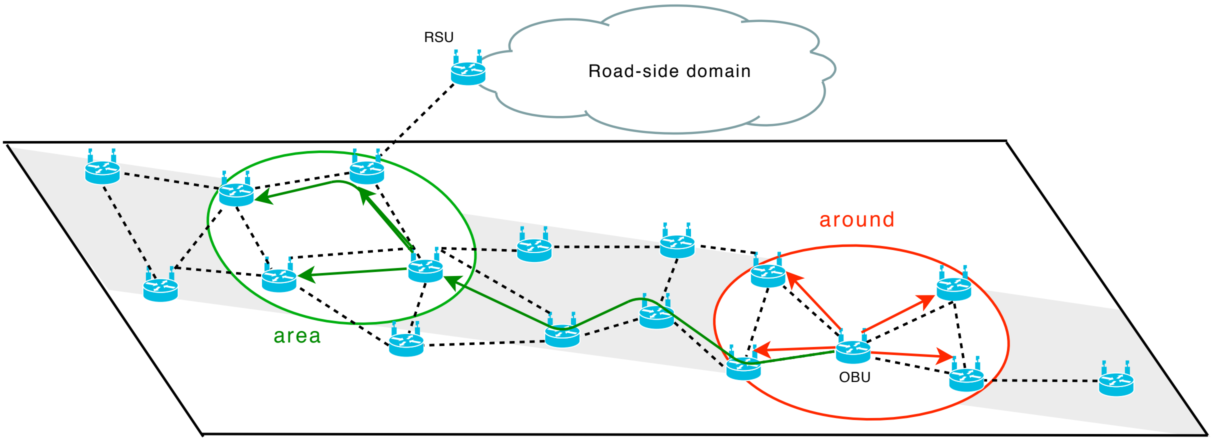


Figure 12: Types of Geo-Broadcast

11.2 Geo-Broadcast

Geo-Broadcast is the communication from one node to all nodes in designed area as illustrated in figure13. Geo-Broadcast has types depending on the scenarios. V2V Geo-Broadcast is communication starts from a single vehicle and ends all vehicles within a certain geographical area(s). I2V Geo-Broadcast is the communication starts from a single point at infrastructure and ends at all vehicles within a certain geographical area(s).

The C2C NET IDs are allocated to the interfaces of OBUs and the C2C area ID is allocated to the geographical areas. The packet for geo-broadcast is designated to the C2C area ID that matches to geographic area (Marked with blue text in figure 13). The destination C2C area ID is actually translated into the set of (latitude, longitude, Radius) and put to the C2C NET header of Geo-broadcast mentioned in figure3.

For V2V Geo-Broadcast, the C2C area ID of 'C2C-9' is specified as the destination of an application in figure 13. The C2C area ID (C2C-9) is translated to the set of (latitude, longitude, Radius) by looking up location table in the OBU. The packets of the geo-broadcast are broadcasted in the geographic area of C2C-7 thanks to the geographic routing functionality of C2C NET.

For I2V Geo-Broadcast, the C2C area ID of 'C2C-7' is specified as the destination of an application in figure 13. The C2C area ID (C2C-7) is translated to the set of (latitude, longitude, Radius) by looking up location table in the OBU. The packets of the geo-broadcast are broadcasted in the geographic area of C2C-7 thanks to the geographic routing functionality of C2C NET.

Around Geo-Broadcast classified in 11.1 is the special version that source OBU specified own C2C NET ID as area ID.

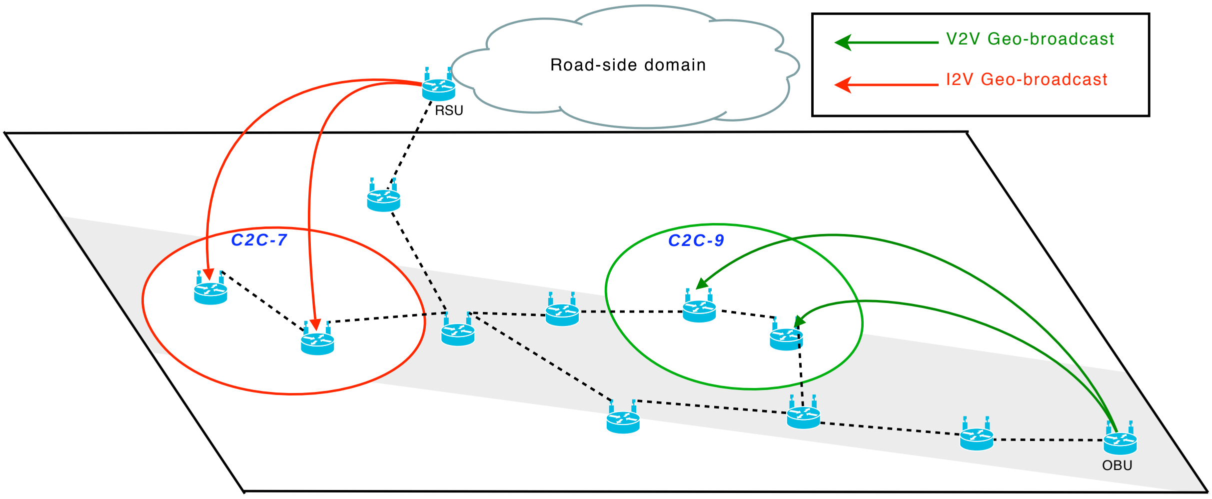


Figure 13: Geocast

11.3 Multicast over Around Geo-Broadcast

Around Geo-broadcast packet delivered within the radius that specified in IPv6 packet. The radius is specified in the IPv6 destination multicast address. The Geonet project has newly specified multicast address as the destination address. Following addresses (Table 4) are proposed, however it is not fixed yet. The address is to be declared by IANA in the future.

The overview of the packet delivery of Around Geo-Broadcast is illustrated in Figure. 14. The example shows that AUs send multicast packet to all the nodes within **500**m from OBU1 by specifying FF0E::**500** as the destination address. The OBU1 put own position to destination node latitude and longitude. Then radius in the C2C NET header is set to 500.

	8	4	4	64	16	32
RFC 2373	11111111	flags	scope	Group ID (96 bits)		
Around	11111111	flags	scope	::	Radius	Group ID
Area	11111111	flags	scope	Area ID	Radius	Group ID

Table 4: Multicast Address specified in Geonet Project (TBA)

C2C NET geographic routing brings the packet to all the nodes in the area from OBU1. All the OBUs that have receive the C2C NET packet checks that there are listeners of the multicast address behind it. When one or more listeners connect to the in-vehicle network, the OBU forwards the multicast packets to the network. The discovery of the listener is performed as MLDv2 (RFC3810 [7]).

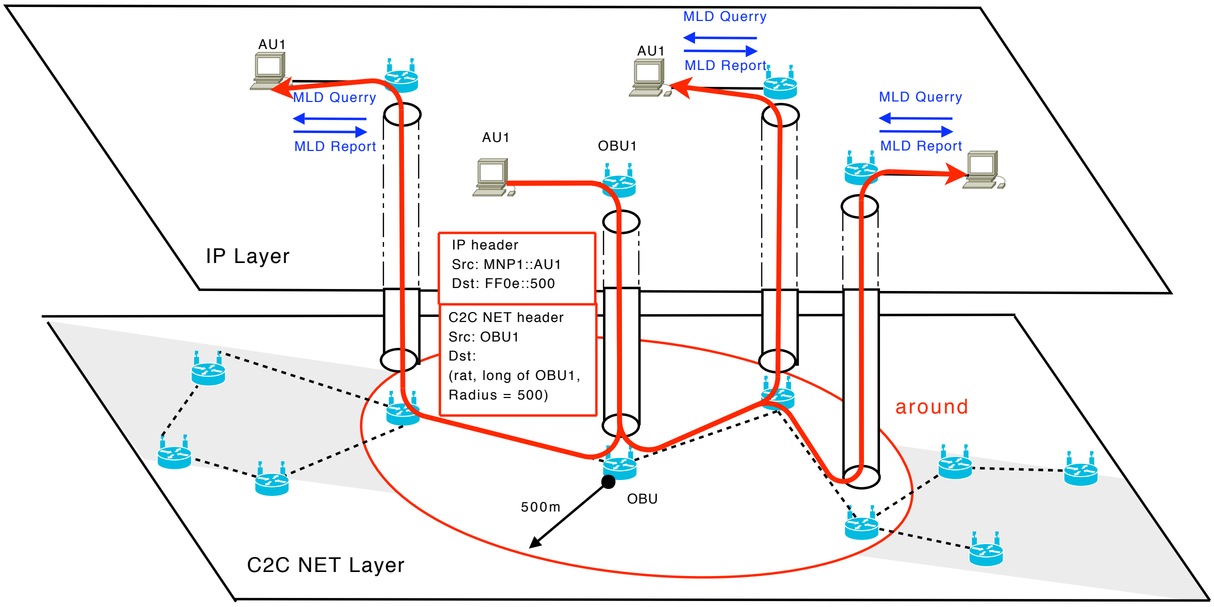


Figure 14: Multicast Over Around Geo-Broadcast

11.4 Multicast over Area Geo-Broadcast

11.4.1 Approaches

In the section, two issues are described and propositions are reported for the issues.

Since the C2C NET layer is below the IPv6 layer, the multi-hop C2C NET link looks a single link at IPv6 point of view. In other words, all of OBUs and RSU are connected to a big hub as illustrated in figure 15.

In the configuration of figure 15, multicast group groups some of AUs (AU2/AU3/AU4 and AU5/AU5). Straightforward approach is using Multicast Listener Discovery (MLD) Protocol among the nodes. However this approach makes additional overhead of signaling in C2C NET link, while the Geo-Broadcast capability is already provided by C2C NET Layer. Thus, IPv6 multicast over Geo-Broadcast should not provide functionality to make multicast group, but only forwarding of the multicast packets.

In IP layer, the multicast packets from the source are broadcasted in the C2C NET link. Then the packets are delivered to appropriate OBUs by C2C NET geo-broadcast functionality. The destination IPv6 multicast address should contain the destination C2C NET ID. Source AU can specify geographic area by specifying IPv6 multicast that contain destination C2C NET ID. The source can be attached to RSU or OBU. The multicast mechanism is common.

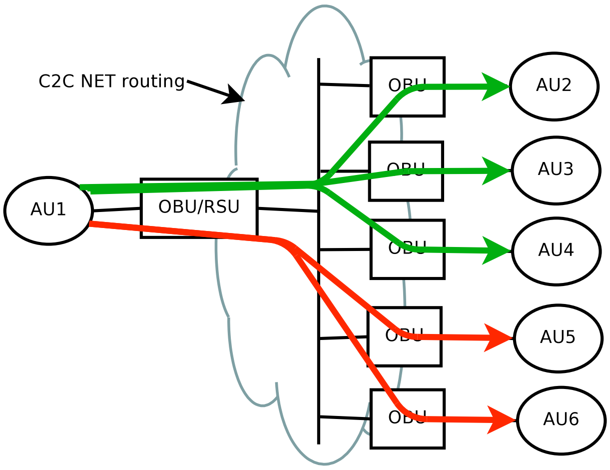


Figure 15: IP model on Geo-Broadcast

Once multicast group are made, the multicast can send the packets to the multicast address. In multicast over C2C Geo-Broadcast, the multicast address is made from multicast prefix and C2C area ID. The C2C area ID means certain geographic area in C2C NET geo-broadcast case. An example is drawn in figure 16.

Two approaches can be considered in the case. First one is orthodox approach that multicast source sends multicast address of the end nodes. In the case, destination IPv6 mutlticast address is FF0e::C2C-area1. 'FF0e::' is allocated to global multicast address and C2C-area1 is the C2C area ID that means certain geographic area. AU1 specify the geographic area by selecting the C2C area ID. The source address of the multicast packet is Prefix::AU1 that is an address of the network AU1 attached. When OBU/RSU makes C2C NET header, it looks destination address of IPv6 header (FF0e::C2C-area1) and takes C2C area ID (C2C-area1) from the header. Then destination address of C2C NET header is made from the C2C area ID.

However the approach has the issue that the listener (AU2) has to know that multicast address to listen the multicast packets. This is not easy for AU2 because the C2C area ID that means certain geographic area (C2C-area1) is changed accordingly to the movement of the vehicle. Since AU2 isn't assumed having GPS receiver, it is difficult for AU2 to listen multicast address (FF0e::C2C-area1).

To solve the issue, the second approach comes up. The multicast address (FF0e::C2C-area1) is used to transfer the packets until OBU2 and OBU2 translates the destination address as FF02::1 (all node multicast). This approach allows AU2 not knowing multicast address ahead. In addition, all the AUs can receive the multicast packets without having any new multicast functionality. The second approach is preferable for IP multicast over Geo-Broadcast.

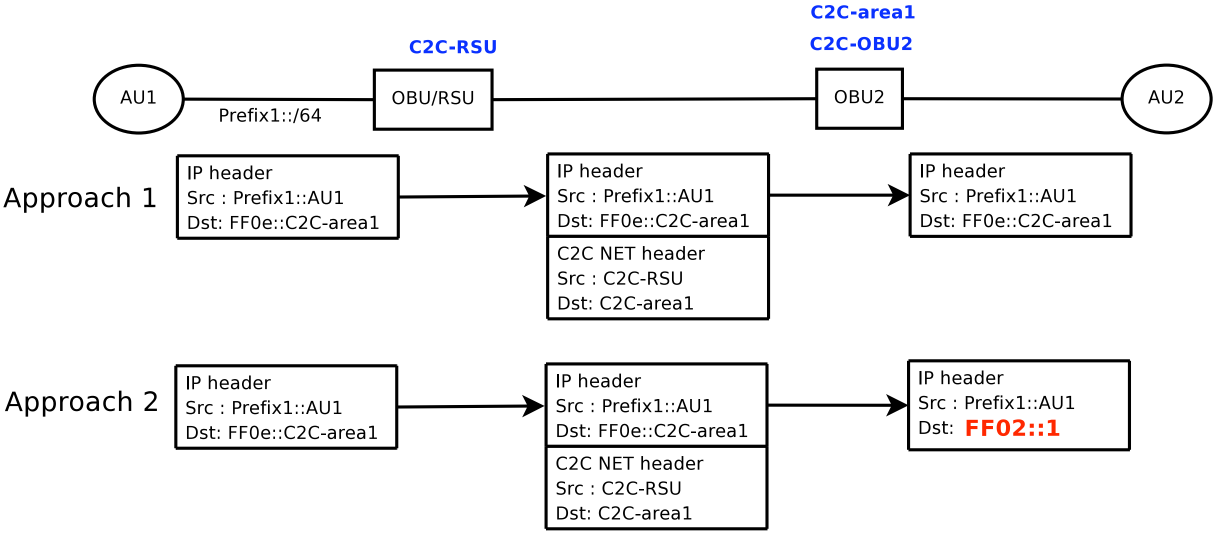


Figure 16: Multicast Grouping

11.4.2 System Overview for IP Multicast over Area Geo-Broadcast

The overview of the IP multicast over Geo-Broadcast is illustrated in figure based on the approach described in 17 section11.4.1. In the figure, IPv6 layer is illustrated above and C2C NET layer is illustrated below. (Note that the figure shows same scenario as figure 13 but with IPv6 support.)

For V2V Geo-broadcast, AU1 sends IPv6 multicast a packet to AUs attached to the vehicles running on the geographic area of C2C area ID 'C2C-9'. Thus the IP destination is set to 'FF0E::C2C9' (and source address is set to MNP1::AU1). The OBU attaches the destination C2C area ID (C2C-9) by taking from destination address of IPv6 header. By C2C NET Geo-Broadcast functionality, the C2C encapsulated packet is delivered to OBUs running in the geographic area of 'C2C-9'. The OBUs transfer the packet inside the vehicles with destination address of all node multicast address (ff02::1). By using all node multicast address, all the AUs connected to the vehicles can receive multicast packet without knowing the multicast group.

I2V Geo-broadcast takes the same approach. AU6 sends multicast packet to the geographic area of 'C2C-7'. The destination IPv6 address is 'FF0E::C2C-7' and source address is Prefix1::AU6. The IPv6 multicast packets are encapsulated by C2C NET header that has C2C-7 as destination C2C area ID and RSU as source C2C NET header. They are brought to OBUs in geographic area of 'C2C-7' as well as the V2V Geo-broadcast case. The OBU sends the multicast packets to all node multicast address in the vehicle network.

12 Router Advertisement

OBU configures Care-of address on its egress interface by Router Advertisement. Router Advertisement on C2C NET uses Geo-Broadcast type of communication. In the section, Geo-broadcast packet and Router Advertisement propagation is detailed.

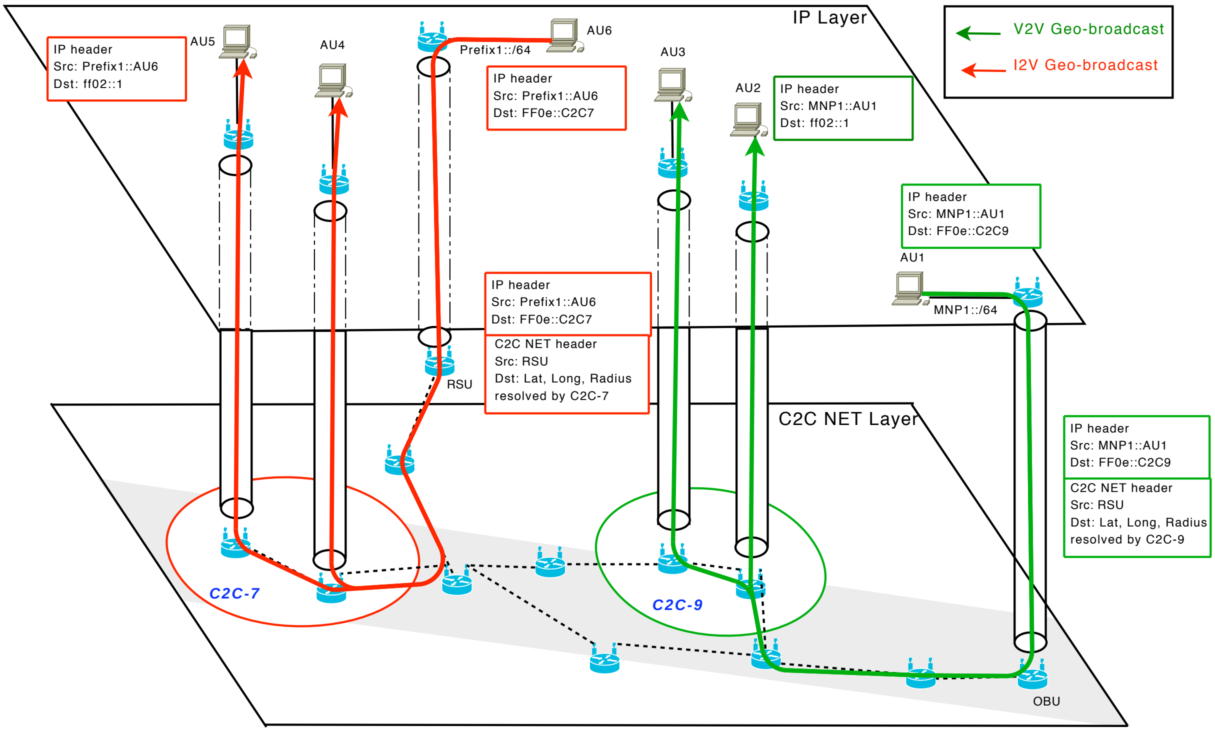


Figure 17: Multicast over Geo-Broadcast

12.1 Router Advertisement Propagation

Figure 18 shows Router Advertisement propagation in C2C NET. The Source address of Router advertisement MUST be the link-local address assigned to the interface from which this message is sent, as specified in [5]. In the case of Figure 18, it is set as fe80::C2CID1. The destination address is typically the Source Address of an invoking Router Solicitation or the all-nodes multicast address (FF02::1). In C2C NET Router advertisement specifies special multicast address as destination address described in section. 11.3 .

According to the IPv6 destination address of RA, C2C NET header is added. Source C2C NET ID is C2C NET ID of the RSU, and the position information (latitude, longitude and attitude) of RSU is added to C2C NET header as well. Radius value of C2C NET header is added according to the destination IPv6 address of RA. When FF0E::**1500** is specified in source address of RA as in figure 18, the value of the radius in C2C NET is set to **1500**m. We are now discussing about dynamic range selection in the RSU. The selection can be performed by selecting the destination address of RA.

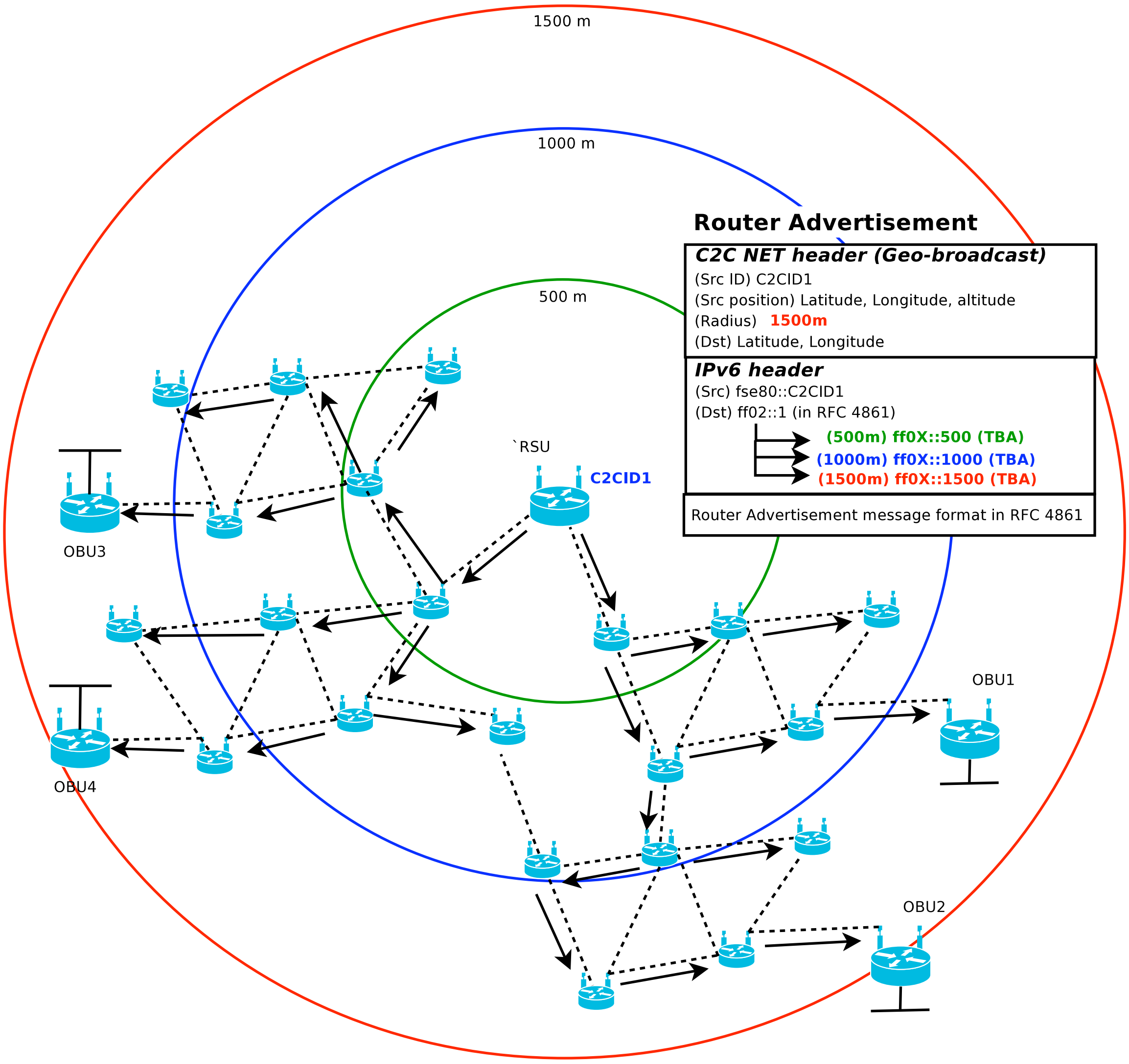


Figure 18: Router Advertisement Propagation

Part IV

IPv6 forwarding & interface management

13 IPv6 interfaces

An MR has at least one ingress interface and egress interface. In geonet, at least one IEEE802.11p interface must be used with C2C NET, but also other interface can be used such as Standard WLAN (IEEE802.11a/b/g) or 3G interfaces. The interface except for IEEE802.11p may be used without C2C NET. Thus routing table of an MR mainly maintains four kind of interfaces as follows. The four interfaces are illustrated as (a), (b), (c) and (d) in Figure 19.

a. C2C interface

b. Other egress interface

c. NEMO tunnel over C2C interface

d. NEMO tunnel over other egress interface

The packets passed to C2C interface (illustrated as (a) in Figure 19) is encapsulated with C2C header after IPv6. And finally they are encapsulated by IEEE802.11p MAC header and actually emitted to the air. Thus C2C interface is recognized as tunnel interface in the kernel.

Other interface (illustrated as (b) in Figure 19) is 'normal' interface. The data is sent from the routing table is encapsulated by MAC header of the link type and emitted to the air.

NEMO tunnel over C2C interface (illustrated as (c) in Figure 19) is tunnel interface. The data is first encapsulated by IPv6 header (source address: Care-of Address, destination address: HA address). Then the data is encapsulated by C2C NET header. Finally the data exit from OBU with MAC header.

NEMO tunnel over other egress interface interface (illustrated as (d) in Figure 19) is tunnel interface. The data is first encapsulated by IPv6 header (source address: Care-of Address, destination address: HA address). Finally the data exit from OBU with MAC header.

14 Routing table setup

To distribute packets to multiple paths simultaneously in the MR, policy routing is used. Classic routing mechanisms are not suitable, because of the 'longest match' principle. We propose to introduce multiple routing tables using Route Policy Database (RPDB) to the system as shown in Figure 20. The RPDB allows to maintain several independent routing tables in the kernel. Each packet can then be routed according to one of these tables. The determination of which routing tables should be used in a particular case is up to the implementer. It is usual to route depending on the type of flow that is being routed.

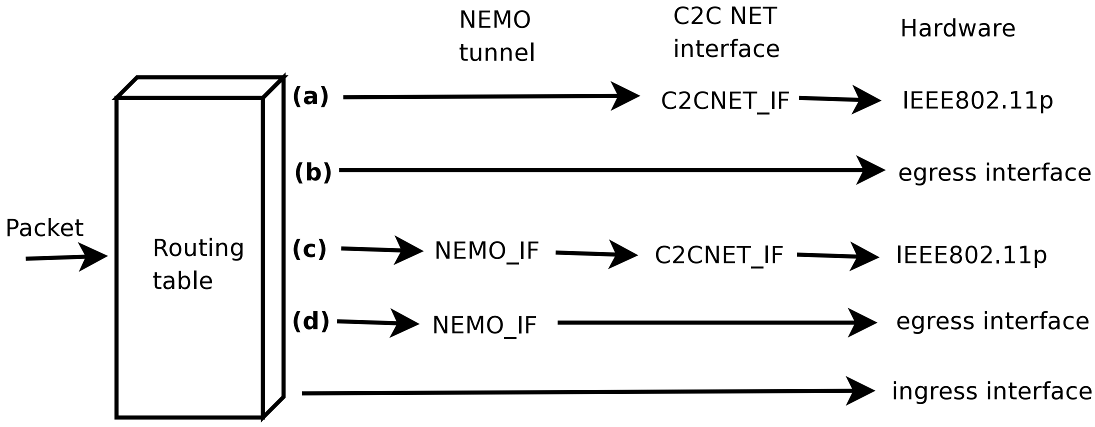


Figure 19: Routing on MR

In IPv6 over C2C NET case, an MR maintains at least five routing tables. (a) C2C_NET is the routing table for C2C NET native packet. The packet is geonet unicast, geonet-anycast or geonet broadcast for V2V communication. (b) normal is the routing table for the packet emitted to the air of Egress interface. The number of the routing table of this type is same number of the egress interfaces equipped in the MR. These routing tables are used for V2V communication. (c) C2C+NEMO is the routing table for NEMO and C2C NET. This is used for V2I communication and finally emitted to IEEE802.11p interface. (d) NEMO routing table is for V2I communication. The number of the routing table of this type is same number of the egress interfaces equipped in the MR.

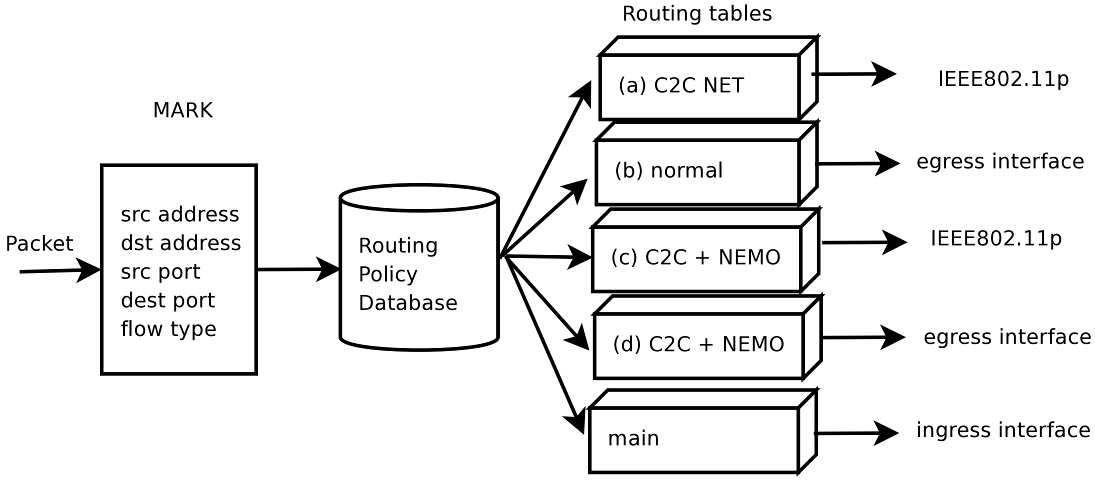


Figure 20: Policy routing

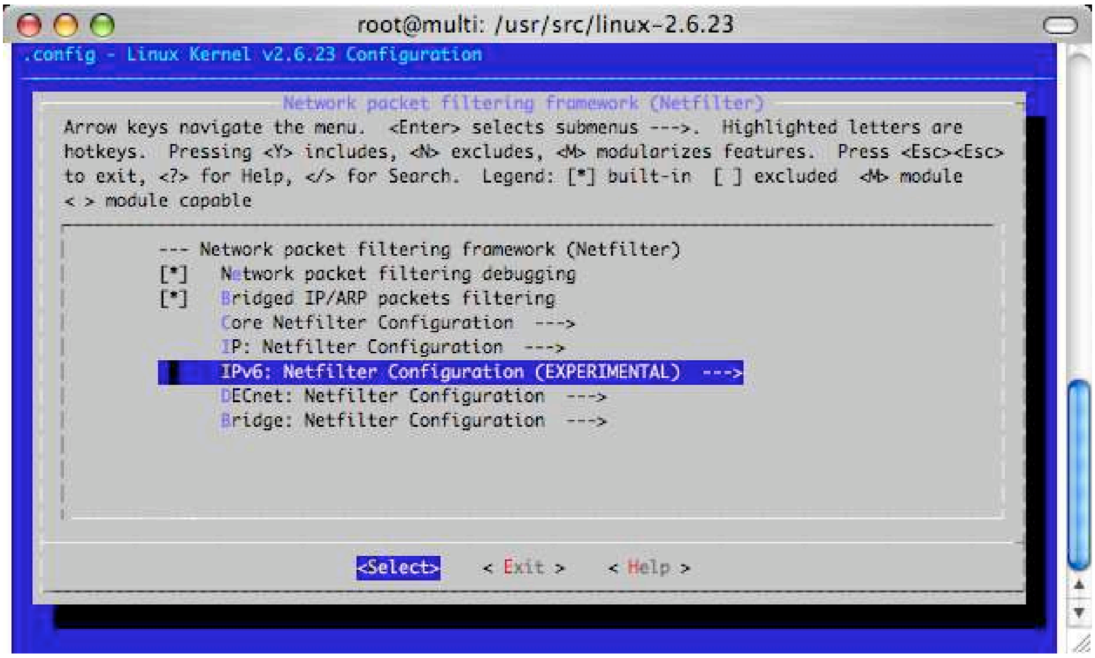
15 Virtual interface between IPv6 and C2C NET

Policy routing is implemented as netfilter [4] in Linux system. We've decided to use tap0 as the interface between IPv6 and C2C NET in Linux system. In the section, usage of the netfilter is mentioned.

15.1 Using policy routing

To use IPv6 netfilter, the kernel option should be set correctly. In Linux kernel version 2.6.23, following option should be set. Figure 21 shows the screen shot of setting kernel option.

```
Networking
–> Networking options
   –> Network packet filtering framework (Netfilter) [CONFIG_NETFILTER]
      –> Core Netfilter Configuration
         –>  Netfilter Xtables support (required for ip_tables) [CONFIG_NETFILTER_XTABLES]
         –> "MARK" target support [CONFIG_NETFILTER_XT_TARGET_MARK]
         –> "mark" match support [CONFIG_NETFILTER_XT_MATCH_MARK]
      –> IPv6: Netfilter Configuration (EXPERIMENTAL)
         –> IP6 tables support (required for filtering) [CONFIG_IP6_NF_IPTABLES]
         –> Packet mangling [CONFIG_IP6_NF_MANGLE]
```



```
root@multi: /usr/src/linux-2.6.23
.config - Linux Kernel v2.6.23 Configuration
               Network packet filtering framework (Netfilter)
 Arrow keys navigate the menu.  <Enter> selects submenus --->.  Highlighted letters are
 hotkeys.  Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc>
 to exit, <?> for Help, </> for Search.  Legend: [*] built-in  [ ] excluded  <M> module
 < > module capable

         --- Network packet filtering framework (Netfilter)
         [*]   Network packet filtering debugging
         [*]   Bridged IP/ARP packets filtering
               Core Netfilter Configuration  --->
               IP: Netfilter Configuration  --->
               IPv6: Netfilter Configuration (EXPERIMENTAL)  --->
               DECnet: Netfilter Configuration  --->
               Bridge: Netfilter Configuration  --->

                     <Select>    < Exit >    < Help >
```

Figure 21: Kernel option for policy routing

15.2 Using tap interface

To use tap0, uml-utilities package is necessary to be installed. If it is not installed, type as follows as super user.

---

[4] Netfilter: `http://www.netfilter.org`

```
# apt-get install uml-utilities
# modprobe tun
# tunctl
# ifconfig tap0 up
```

15.3 Filtering the packets

Figure 22 shows the entire procedure that the packets from MNNs are forwarded to next router from the MR. The steps are as follows: (1) Filter the packet by source address, destination address, source port, destination port and flow type, (2) set the rule of the policy, (3) configure the each routing table.

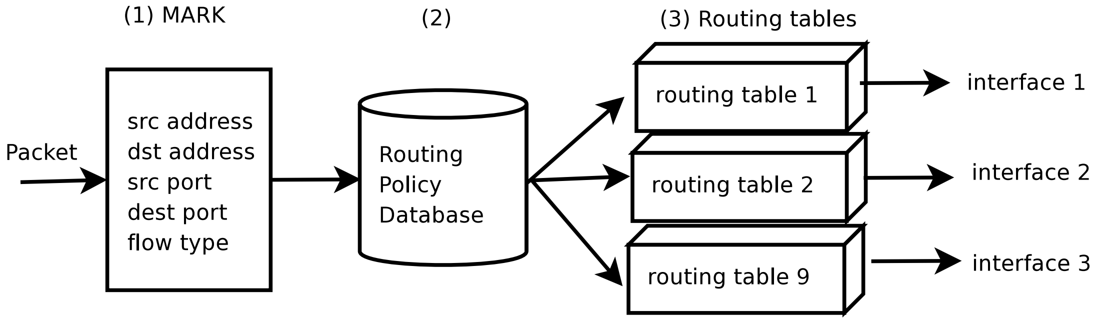


Figure 22: ip6tables

First, the packet is filtered by various metrics (source address, destination address, source port, destination port and flow type,) and then marked by number ((3) in Figure 22). For example, the first one of following command filters ICMPv6 packet from 2001:1000:2000: 3001::3 (source address) is marked with 9. The second one filters UDP packet distained to port 2000 is marked as 1.

```
# ip6tables -A PREROUTING -t mangle -j MARK
            -p icmpv6 --source 2001:1000:2000:3001::3
            -j MARK --set-mark 9
# ip6tables -A PREROUTING -t mangle -j MARK
            -p udp --dport 2000
            -j MARK --set-mark 1
```

Following line flush the all filter rules.

```
# ip6tables -F
```

15.4 Adding policy rule

Second, the packet marked by the number like in Section 15.3 is passed to policy rules ((2) in Figure 22). Following example shows the case that the packet marked as 9 comes from Mobile Network Prefix lookups the routing table number 9. Each rule has priority and the command specify the priority of the rules as 301.

```
# ip -6 rule add from 2001:1000:2000:3000::/64
        fwmark 0x9 lookup 9 prio 301
```

If you want to check the rules in the policy, you can check with following line.

```
# ip -6 rule
```

15.5 Modifying Routing tables

Third, the packet sent to each routing table is examined by entries in the routing table ((3) in Figure 22). Following line is the example to add a routing entry to routing table number 9. It add default route from Mobile Network Prefix (2001:1000:2000:3000::/64) to tap0

```
#ip -6 route add default from 2001:1000:2000:3000::/64 dev tap0
       table 9 metric 10 proto 16
```

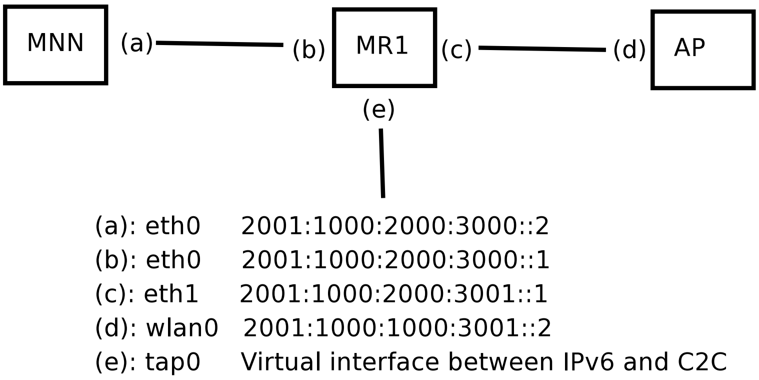
If you want to check routing entries of specific routing table, you can check with following command.

```
# ip -6 route list table 9
```

16 Pre-experiment for IPv6 over C2C NET implementation

16.1 C2C NET interface test

The test used linux PCs. Figure 23 shows the configuration of the test. MR1 equips two interfaces: egress interface(b) and ingress interface (c). MR1 also has the Virtual interface between IPv6-C2C as tap0 that is created by uml-utilities package installed in Section 15.2.



(a): eth0 2001:1000:2000:3000::2
(b): eth0 2001:1000:2000:3000::1
(c): eth1 2001:1000:2000:3001::1
(d): wlan0 2001:1000:1000:3001::2
(e): tap0 Virtual interface between IPv6 and C2C

Figure 23: Test topology

Actual configuration of MR1 is as follows.

```
# ip -6 rule add from 2001:1000:2000:3000::/64
        fwmark 0x9 lookup 9 prio 301
# ip -6 route add default from 2001:1000:2000:3000::/64 dev tap0
# ip6tables -t mangle -F PREROUTING
# ip6tables -A PREROUTING -t mangle -j MARK --set-mark 9
```

After the configuration all the packet from 2001:1000:2000:3000::/64 are directed to tap0 (virtual interface of C2C NET). For example, the ICMPv6 packets from MNN are directed to tap0 interfaces. The packet transmit to tap0 can be seen by tcpdump.

16.2 Example of NEMO MR configuration

Once the C2C NET tap0 interface configured, we can specify the interface as egress interface of mobile router like as follow. The configuration is located in /etc/mip6d.conf. See “NEPL (NEMO Platform for Linux) HOWTO” page [5] for more detail.

```
   NodeConfig MN;
   DebugLevel 10;
   DoRouteOptimizationCN disabled;
   DoRouteOptimizationMN disabled;
   SendMobPfxSols enabled;
   UseCnBuAck disabled;
   # We use Explicit Mode
   MobRtrUseExplicitMode enabled;
   OptimisticHandoff enabled;
   # The Binding Lifetime, for example 20 seconds
   MnMaxHaBindingLife 20;
   Interface "tap0"{
     Bid 200;
     BidPriority 10;
     Reliable true;
   }
   Interface "eth2" {
 Bid 100;
  BidPriority 20;
Reliable true;
   }
   # Replace eth0 with your egress interface
    MnHomeLink "eth0" {
        IsMobRtr enabled;
        HomeAgentAddress 2001:a:b:0::1000;
        HomeAddress 2001:a:b:0::1/64 (2001:a:b:1::/64);
        RegMultipleCoA enabled;
        IfMultipleCoA "tap0", "eth2";
   }
   # IPsec configuration - NO IPSEC AT THE MOMENT
```

---

[5] `http://www.nautilus6.org/doc/tc-nepl-howto-20060209-KuntzR/nepl-howto.html`

```
UseMnHaIPsec disabled;
KeyMngMobCapability disabled;
# EOF
```

17 Conclusion

In the second part of the document, specification and implementation for IPv6 over C2C NET are described. The Router Advertisement propagation on C2C NET uses geo-broadcast packet of C2C NET. The multicast addresses are newly specified in the part of document. Interface between IPv6 and C2C are described. The C2C interface is realized as tap0 interface in Linux system. To use multiple interfaces simultaneously, policy routing is introduced. The policy routing is implemented as netfilter in Linux, and we described the configuration of the policy routing in the document. Pre-experiment is reported.

Part V

IP Multicast over Geo-Broadcast

18 IPv6 address and C2C ID mapping

18.1 Unicast

- The IPv6 address is formed by a network prefix and a C2C NET ID. The following example illustrates how to embed C2C NET ID into IPv6 address.

```
Example:
IPv6 address = 2001:660:3013:3:1234:5678:9ABC:DEF0
C2C NET ID   = 12-34-56-78-9A-BC-DE-F0
```

18.2 Multicast (Anycast)

- The multicast IPv6 address is formed by a network prefix and an areaID. The following example illustrates how to embed areaID into IPv6 address.

```
Example:
IPv6 address = FF3E:40:2001:660:3007:123:1234:5678
areaID    = 12-34-56-78
```

Take notice that the size of areaID is only limited by the number of bits in multicast address.

19

Part VI

Advance Features of IPv6 Over C2C NET

20 Introduction

To support communication in mobile environment, Network Mobility (NEMO Basic support, or NEMO for short [2]) has been standardized in IETF. NEMO Working Group and recommended by the ISO TC204 WG16 draft standard (called CALM: Communications Architecture for Land Mobile environment) to achieve Internet mobility for vehicles.

NEMO is one of key technologies of vehicle communication, however, issues related to Route Optimization still remain in NEMO Basic Support, while they already have been solved in Mobile IPv6 [8]. In NEMO, all the packets to and from MNNs must be encapsulated with IPs in the tunnel between the MR and the HA. Thus all these packets between MNNs and CNs must go through the HA. This causes various problems and performance degradation.

21 Route Optimization

These sub-optimal effects are described as follows. **Suboptimal routes** are caused by the packets being forced to pass by the HA. This leads to increased delay that is undesirable for applications such as real-time multimedia streaming. **Packet Encapsulation** of additional 40 bytes header increases packets overhead and risks of packet fragmentation. This results in an increased processing delay for every packets being encapsulated and decapsulated in both the MR and the HA. **Bottlenecks in the HA** are a severe issue because significant traffic to and from MNNs is aggregated in the HA when it supports several MRs acting as gateways for several MNNs. This may cause congestion at the HA that would lead to additional packet delays, or even packet losses. **Nested Mobile Networks** is an issue that NEMO Basic Support raises by having arbitrary levels of nesting of mobile networks. This permits an MR to host other MRs in its mobile network. With nested mobile networks, the use of NEMO further amplifies the sub-optimality listed above.

In IETF, the issues of Route Optimization of NEMO are addressed in [9]and the solution space is analyzed in [10]. Requirements of Route Optimization in various scenarios are described for networks for vehicles [11] and aeronautic environments [12].

In C2C NET, route optimization scenarios are divided into four scenarios as illustrated in Figure 24. The classification, first, depends on wither the correspondent node (CN) is fixed in the Internet or it connects behind an OBU. When CN is fixed in the internet, two configurations are considered depending on wither it connects to same road-side domain, which the source OBU connects (Scenario 1) or it connects the other network in the Internet (Scenario 2). When correspondent node connects in-vehicle network, which the case that CN is AU, two sub-scenarios are considered depending on wither the OBUs which the CN connects are connecting same C2C NET as the source OBU (Scenario 3) or the CN's OBU is connecting to the other C2C NET than Source OBU (Scenario 4).

Related works are summarized in Appendix. The related mobile technologies are described in section 25.1: Mobile IPv6, Route optimization in Mobile IPv6, NEMO and Multiple Care-of Address Registration. Then, existing propositions for Route Optimization in NEMO are classified into five types in section 25.2. Since scenarios for C2C NET is mainly non-nested case, non-nested solution would help to have an idea for the solution such as (a) Binding Management on Correspondent Entity, (b) Infrastructure-based Route Optimization and (c) Route Optimization using MANET.

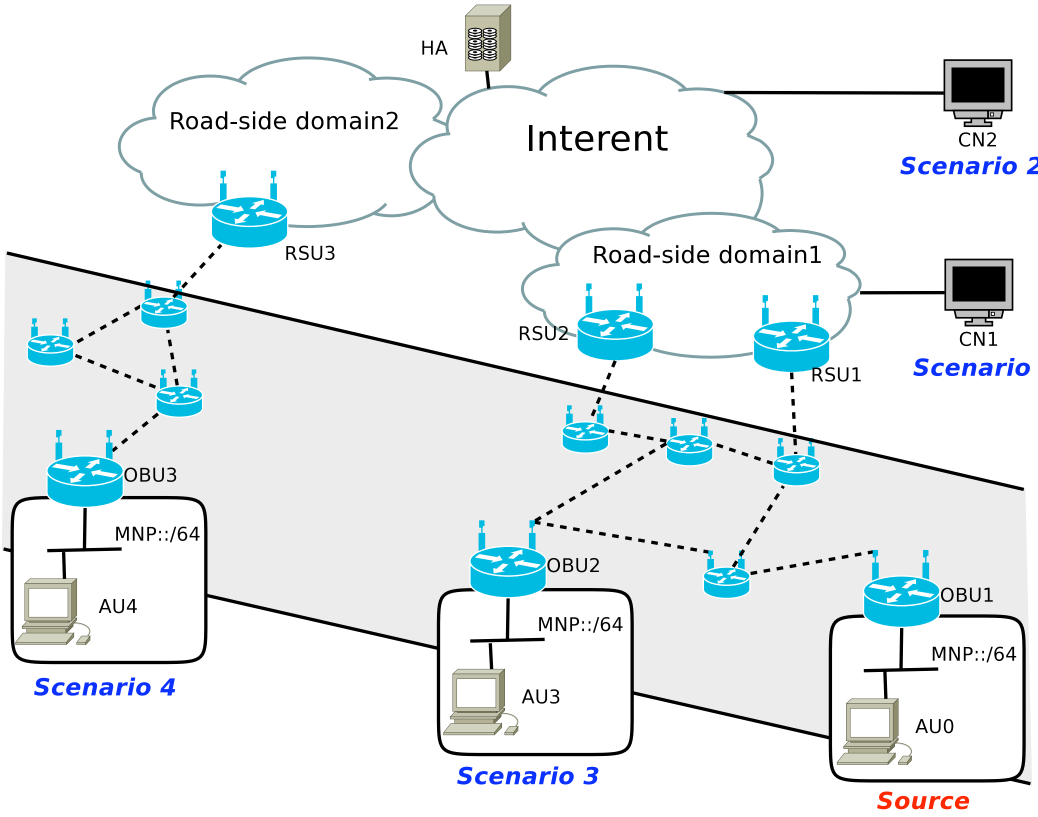


Figure 24: Scenarios

22 Requirements

To propose a solution for route optimization in the four scenarios, five requirements are mentioned in the section.

- **Maximum optimization**
  First meaning of route optimization is bypassing HA in some way, however the sub-optimal effects vary in several reasons such as sub-optimal routes, packet encapsulation, bottleneck in the HA and nested mobile networks as described in section 21. If the solution could not solve all suboptimal effect listed above in some scenarios, the solution should optimize the effect as much as possible. By definition of route optimization, the solution must bypass the HA. On the other hand, packet encapsulation may be used when there is no other way to avoid the encapsulation.

- **Supporting all scenarios**
  The solution must support all the scenarios mentioned in section 21.

- **No additional functionality for end-node**
  In C2C NET, end nodes are either MNN or CN. NEMO allows the MNN not to have mobility management functionality and keep them as IPv6 standard node. The route optimization solution should not break this advantage of NEMO, thus the solution let the AU as an IPv6 node. CN side is also expected as standard IPv6 nodes such as existing IPv6 server. The solution must not require additional functionality to both of MNN and CN for deploying the solution easily.

- **Optimization policy installation**
  Route optimization makes the vehicle communication performance better than the communication via HAs. However the route change and signaling for the route optimization makes some disconnect time or some overhead. Thus users or operator should be able to decide if the route optimization is necessary depending on various information. ex. Destination and source address, destination and source port, flow type, duration of traffic, network performance, position and movement of the vehicle.

- **Security consideration**
  The solution must keep relevant level of security from NEMO basic support and C2C NET.

23 Approach

To solve the issues listed in section 21, first we determine the approach for NEMO route optimization. The approach is summarized in figure 25. According to the comparison in section 25.2, (c) Route Optimization using MANET can escape from all the issues such as longer route, packet encapsulation, bottleneck in the HA. By this meaning, the group of the solutions is the best. However the solutions have a limitation of area which the routes are propagated. Hence the solutions only works when there are alternative route to the destination. Thus the alternative routes should be used as far as the route exists for the destination in certain area (Scenario 1 and Scenario 3 in figure 25).

If the alternative route is not exchanged between source and destination, Binding Cache transfer approach is taken as the second best. There are two Binding Cache transfer approaches as listed in section 25.2 as non-nested case, that are (a) Binding Management on Correspondent Entities and (b) Infrastructure-based Route Optimization. Former approach transfers the Binding Cache to the other end from MR, while the later transfer it to intermediate nodes.

Since the object of the study is maximizing the performance vehicular communication, it is preferable to control the entire path between vehicles. Thus the Binding Cache of the MR should be transferred to the other end side of the path as far as possible. Usually, the communication ends between vehicles are MNNs, but an MNN does not have mobility functionalities as designed in NEMO, MNN should not have any mobility functionality. Thus we take the approach to transfer Binding Cache of MR to the other side of MR.

In this case, route optimization between OBU and CN in the Internet are by establishing bi-directional tunnel between OBU and Correspondent Router (CR) near CN in order to keeping the CN as IPv6 standard node (Scenario 2). Route optimization between OBUs is performed by establishing bi-directional tunnel directly between OBUs (Scenario 4). The approaches are classified in table 5.

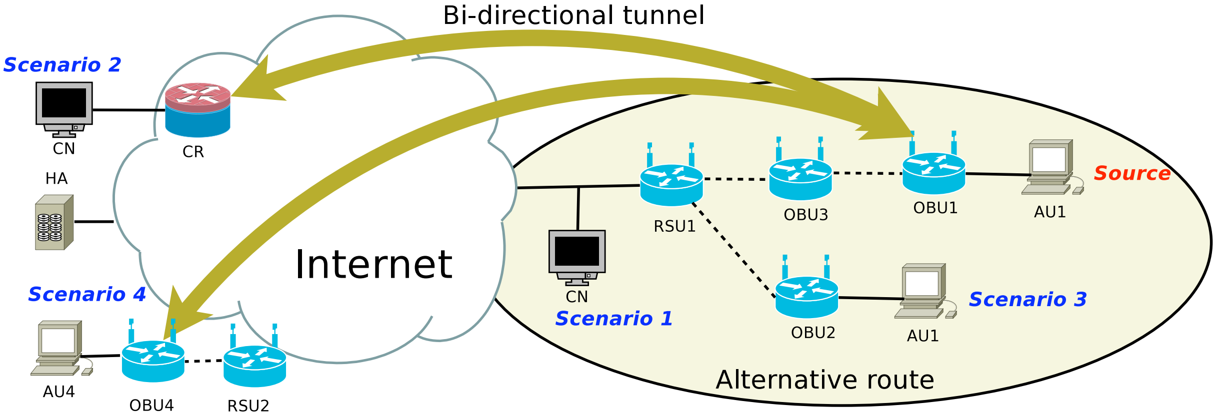


Figure 25: Two modes route optimization

		CN position	
		In fixed network	In-vehicle network
Alternative route	exist	Scenario 1	Scenario 3
	not exist	Scenario 2	Scenario 4

Table 5: Approaches

24 System Overview

In the section, the proposed section is mentioned. The system is classified into two modes as described in section 23 :

- Using alternative route (Scenario 1 and Scenario 3)
- Using tunnel (Scenario 2 and Scenario 4)

When AU in the vehicle starts communication, the OBU decides which mode should be used by checking if the alternative route are entered in the routing table. If there is no entry to the destination except for default route, the OBU should use the second mode.

24.1 Using alternative route

24.1.1 Between OBU and CR (Scenario 1)

Figure 26 shows the example for using alternative route between OBU1 and CR (Scenario 1). OBU1 connects to C2C NET and RSU1 interconnects with both the C2C NET and Roadside domain. In the routes of roadside domain are configured by dynamic routing protocol.

OBU1 receives RA of RSU1 and configures the egress interface as Prefix1::OBU1. At the same time, OBU1 announces in-vehicle network prefix (MNP::/64) to the C2C NET with the method discussed in Part II. Thus all the nodes within the C2C NET know that the in-vehicle network prefix is connected behind OBU1. Once RSU1 receives the announcement, RSU1 propagates the in-vehicle network (MNP::/64) in roadside domain. The CR near the CN receives the announcement and learns the position of the in-vehicle network. By this way, the route from CN to AU1 is configured.

On the other hand, the prefix that CN is attached (Prefix3::/64) is exchanged in roadside domain by dynamic routing protocol and RSU announces the prefix into the C2C NET. By this way, the OBU1 learn the route to the prefix that CN is attached (Prefix3::/64). When there are a lot of prefixes in the roadside domain, RSUs have to announce all of them to C2C NET with maximum aggregation of the routes. By this way, the route from AU1 to CN is setup. The routing entries of CR and OBU are listed in figure 26.

When AU1 starts communication with CN, OBU1 has the routing entry to Prefix3::/64 in this case. Packet is encapsulated in C2C NET header and delivered to RSU1 by C2C NET. Not that NEMO tunnel is not necessary and the communication is free from packet encapsulation. RSU1 to CN is delivered according to the route exchanged by dynamic routing protocol. Returning route from CN to AU1 is also exchanged by dynamic routing protocol. The packet is delivered from CN to RSU1 and emitted from RSU1 with C2C NET header encapsulation. RSU1 set OBU1's C2C NET ID to C2C header by taking from IPv6 address on OBU's egress interface (Prefix1::OBU1).

When OBU1 moves under RSU2, the route between CN and AU1 should be immediately updated. The RSU2 announces the access prefix of CN (Prefix3::/64) as well as RSU1, OBU1 can learn the new gateway to access CN. The route advertisement of Prefix3::/64 should not mixed in C2C NET thanks to the area border realized by geographic routing functionality of C2C NET. The C2C NET doesn't advertise the prefixes within roadside domain beyond the area border. OBU1 also changes the source address of the advertisement to the IPv6 address generated from new prefix (Prefix2::OBU1). The announcement arrives to CR by dynamic routing protocol executed in roadside domain. The CR's routing entry to MNP::/64 are updated to Prefix2::OBU1.

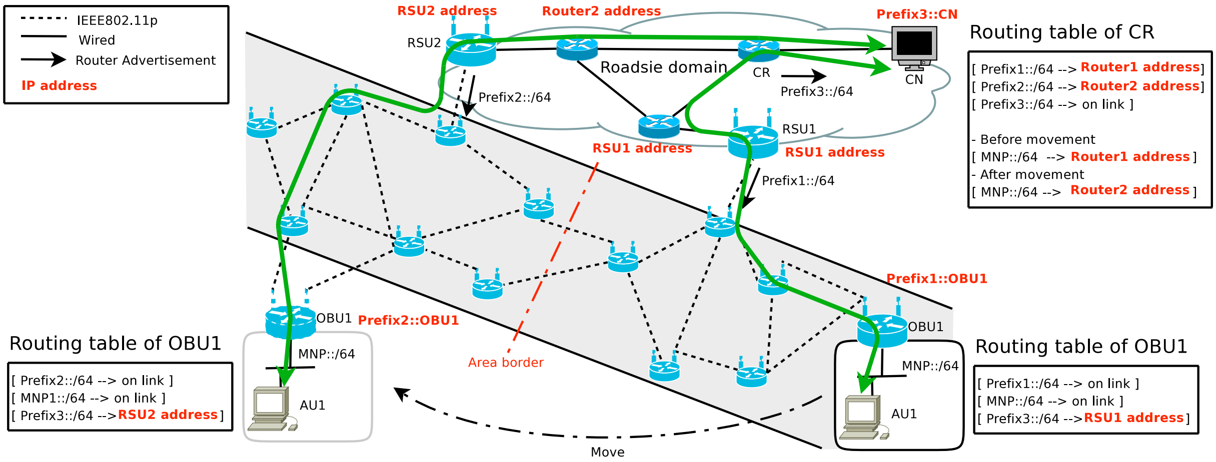


Figure 26: Using alternative route between OBU and CR (Scenario 1)

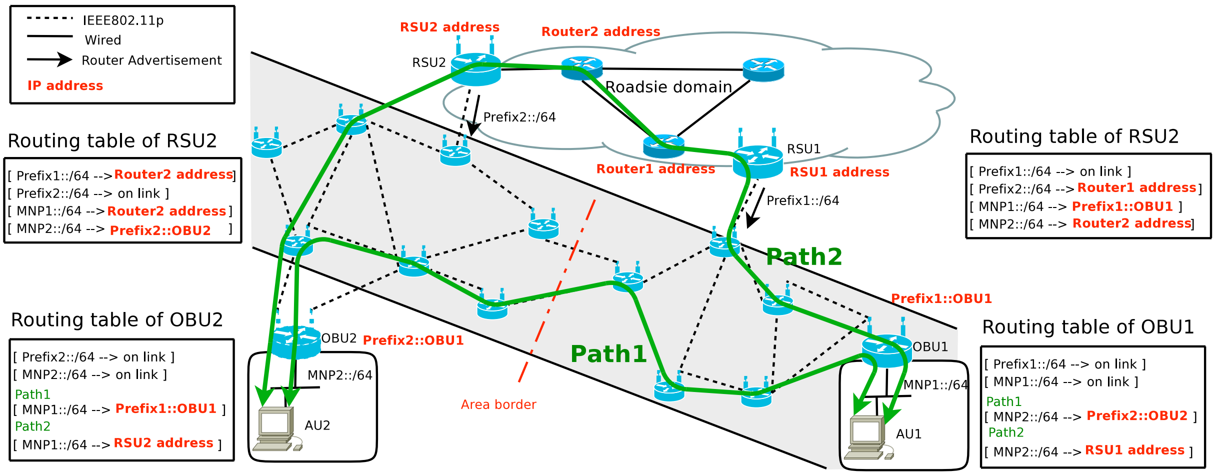
24.1.2 Between OBU and CR (Scenario 3)

Figure 27 shows the example for using alternative route between OBUs (Scenario 3). RSU1 and RSU2 send RAs (Prefix1 and Prefix2, respectively) into C2C NET. In the example, OBU1 and OBU2 are connected to different RSUs (RSU1 and RSU2, respectively). Thus OBU1 and OBU2 configure IPv6 addresses from different prefixes. (If the OBUs are connected to same RSU, the OBUs configure IPv6 addresses from the same prefix. In this case, OBUs can exchange routes to in-vehicle network each other by the way discussed in part II.)

When source and destination OBU are connected to different RSU, two paths can be considered as marked as Path1 and Path2 in figure 27. The two paths cannot used simultaneously, because the routing entry for the paths are overwritten each other in the OBUs' routing tables. Example of routing table is illustrated in figure27.

Path1 is established by exchanging route to in-vehicle network each other between source and destination OBUs. The messages to exchange the routes are same when OBUs are connected in same RSU, but the messages are brought further beyond the area border.

On the other hand, Path2 is established by exchanging routes between OBUs via roadside domain. RSU1 receives in-vehicle network prefix announcement from OBU1 and forwards it to roadside domain. RSU2 receives it and forward to the C2C NET. OBU2 receives the OBU1's in-vehicle announcement from RSU2 and learn the route to MNP::1/64 is via RSU2. Returning route of Path2 is propagated by the same way.



Routing table of RSU2

[Prefix1::/64 --> **Router2 address**]
[Prefix2::/64 --> on link]
[MNP1::/64 --> **Router2 address**]
[MNP2::/64 --> **Prefix2::OBU2**]

Routing table of OBU2

[Prefix2::/64 --> on link]
[MNP2::/64 --> on link]
Path1
[MNP1::/64 --> **Prefix1::OBU1**]
Path2
[MNP1::/64 --> **RSU2 address**]

Routing table of RSU2

[Prefix1::/64 --> on link]
[Prefix2::/64 --> **Router1 address**]
[MNP1::/64 --> **Prefix1::OBU1**]
[MNP2::/64 --> **Router2 address**]

Routing table of OBU1

[Prefix1::/64 --> on link]
[MNP1::/64 --> on link]
Path1
[MNP2::/64 --> **Prefix2::OBU2**]
Path2
[MNP2::/64 --> **RSU1 address**]

Figure 27: Using alternative route between OBU and CR (Scenario 3)

24.2 Using tunnel

When OBU doesn't have routing entry to destination except for default route, the packet goes trough the NEMO tunnel. The packets should be transmitted to the HA. The packets should be transmitted to the HA. After the route optimization, the tunnel between OBUs, and OBU-CR is established directly shown in figure 28. When an AU starts communication with CN or other AU, the tunnel is used. The OBU first decides if the route optimization is necessary, and then discovers the correspondent router (CR). The process is common in the OBU-to-CR route optimization (Scenario 2) and the OBU-to-OBU route optimization (Scenario 4). However the binding management is performed different way depending on the scenarios.

In the section, first Trigger of route optimization and correspondent router discovery are mentioned. Then binding management is described in both Scenario 2 and Scenario 4.

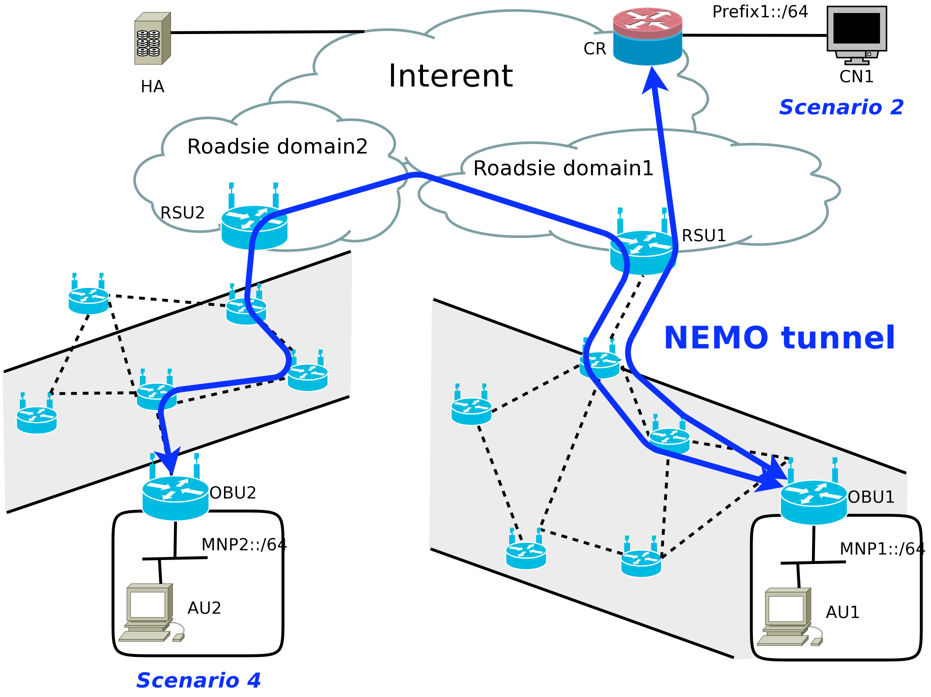


Figure 28: Using tunnel (Scenario 2 and 4)

24.2.1 Trigger of Route Optimization

Once the sub-optimal route via NEMO tunnel is detected by the OBU, the OBU tries to perform route optimization. In the case, both of the OBU and CR (this can be also OBU) should agree with route optimization.

Both sides of OBU potentially can detect the sub-optimal route in Scenario 4. It can be detected by sending packets to the bidirectional tunnel to the HA at sender side OBU. On the other hand, it can be detected by receiving the packets from the bidirectional tunnel from the HA at receiver side OBU. However, suboptimal route detection on sender side OBU has the risk that another OBU doesn't detect the suboptimal route. The route optimization needs to be done with suboptimal route detection of both side OBUs. Thus receiver side suboptimal detection triggers the route optimization initialization. The OBU may have administrative policies to determine if optimized route is necessary for specific traffic. For example, the policy can be based on following factors.

- Destination address, source address
- Destination port, source port

- Flow type
- Duration of the traffic
- Type of Used Egress interface
- Network performance (RTT, bandwidth, packet loss, jitter, etc)
- Position and movement of the vehicle

The policies may differ between two OBUs. Therefore the last OBU should be able to also determine if route optimization should be done. The last OBU can refuse the proposition of route optimization from the first OBU.

24.2.2 Correspondent Mobile Router Discovery

When the MR decides to initialize route optimization with the CR, however, the OBUs initially don't know each other not likely as Mobile IPv6. Communication source and destination are not entities to manage the mobility in NEMO, while communication source and destination manage mobility in Mobile IPv6. Thus correspondant router discovery is necessary.

The procedure is illustrated in figure 29. When OBU1 starts route optimization to OBU1, OBU1 should have signaling with the CR. To discover CR, OBU1 send to CR address discovery request message to the anycast address of AU2's network. The message should be transmitted via basic NEMO route. OBU2 has an anycast address of AU2's network and reply the message of (2) CR reply to OBU1. The reply message includes OBU2's address and OBU1 then knows how to access the OBU from this reply message. After the CR address discovery procedure, each OBU knows the Home address of other OBU.

24.2.3 Binding Management

After OBU1 knows CR address in Scenario 2, it sends binding update to the CR address with NEMO prefix option. The CR replies binding acknowledgement with the prefix managed by CR and configures the tunnel to the OBU's care-of address for the NEMO prefix. The OBU configures the NEMO tunnel to CR address for the prefix managed by CR.

In Scenario 4, both communication source and destination are behind the OBUs which are mobile. In NEMO basic support, the tunnel end can be change by handover. This is the reason why the Binding Update message is necessary. In this case, frequent change of the tunnel end is always OBU side, because of movement of the OBU. The Binding Update message, thus, is sent from OBU to HA. However, in the case of the tunnel between OBUs, the tunnel end can change frequently in both sides of OBUs. This is the necessity of Bidirectional binding update which is illustrated in figure 29 (marked as (3), (4), (5) and (6)).

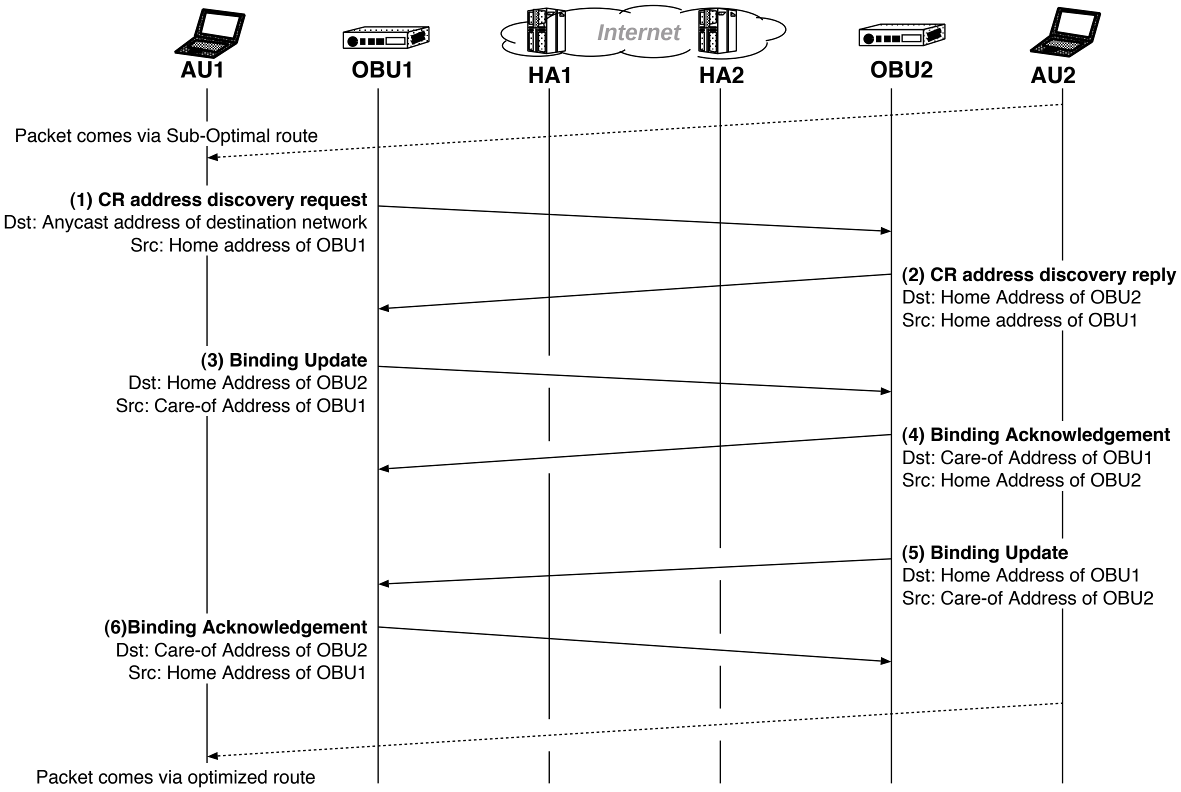


Figure 29: Overview of Route Optimization

25 Appendix (Related Works)

25.1 Mobility Technologies

25.1.1 Mobile IPv6

Mobile IPv6 is the host mobility support protocol standardized as RFC 3775[8] in IETF. The protocl allows a *Mobile Node (MN)* to change the point of attachement one link to another with uninterruped Internet connectivity. MN can connect to the Internet with a permanent and unchanged address named *Home Address (HoA).* A node communicating with MN is called *Correspondent Node (CN)* and packets destinated from CN to HoA are delivered to the *home link* of the MN by normal Internet routing. A fixed router called *Home Agent (HA)* captures the packets and forward them to the current point of attachement of MN by IP-in-IP tunnel. An MN configures an address called *Care-of Address (CoA)* in *foreign link* and always nortifies the address to the HA. The HoA, the CoA and lifetime are included in the notification message called *Binding Update (BU).* The HA receives the BU and replies *Binding Acknowledgement (BA)* to nortifies the acceptance of the BU to the MN. An HA has conceptual data base called *Binding Cache (BC)* to maintain the binding between HoA and CoA. The entries of BC are updated by BU and BA signaling, or expire by the lifetime. MN, on the other hand, maintains Binding Update List (BUL) to store the address of the nodes that maintain BC. An HA address is added to the BUL by default, and CN addresses are added by route optimization described in section 25.1.2. The signaling messages between nodes and Binding database are illustrated in figure 30.

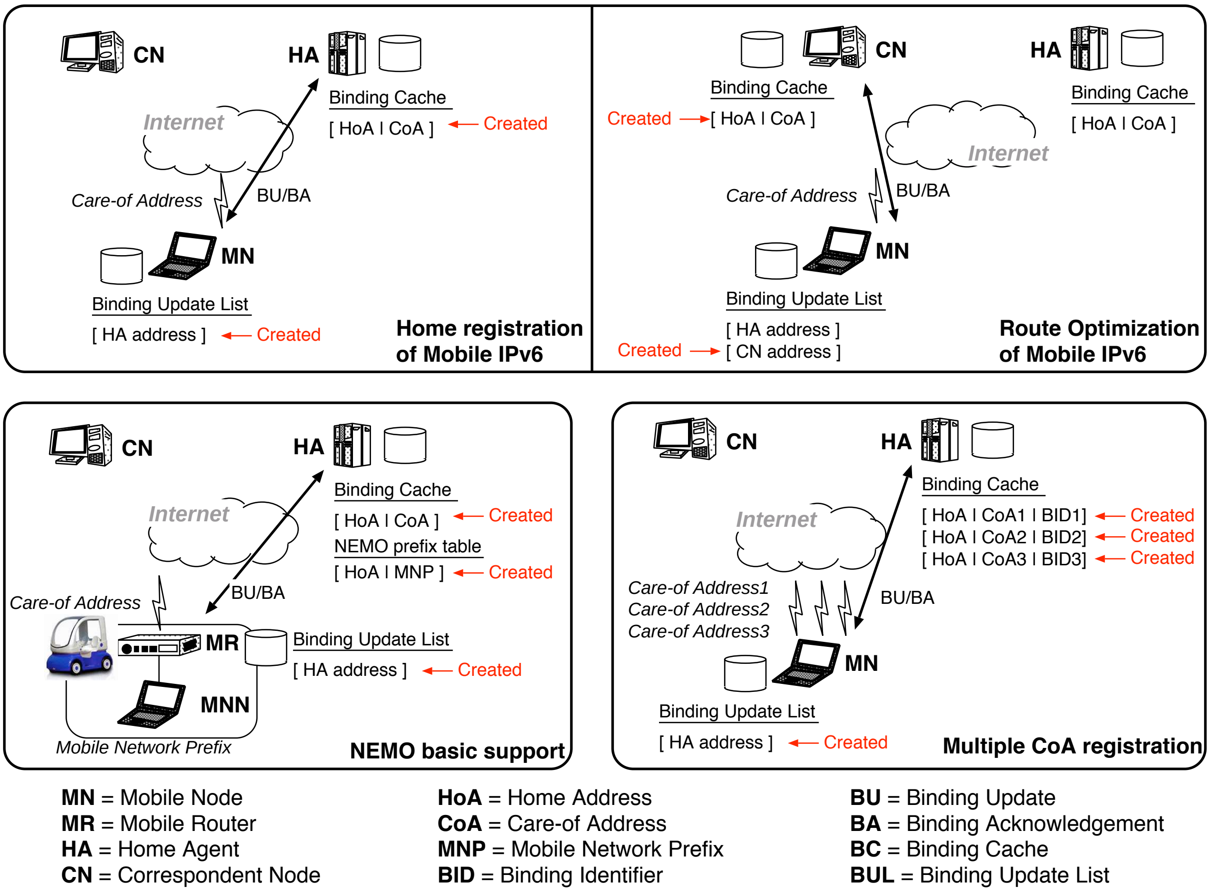


Figure 30: Binding Management of Mobility technologies

25.1.2 Route Optimization in Mobile IPv6

In Mobile IPv6, all the packet between a CN and an MN goes through the HA that supports MN's mobility. To avoid suboptimal route, RFC 3775 define a Route Optimization mechanism to solve the problem. The solution to establish direct path between CN and MN with mobility is illustrated in figure 30. After the communication starts between MN and CN, the MN can try to move the binding chache from the HA to the CN. The MN sends Binding Update to the CN after Return Routability procedure. The procedure guarantees that the binding between the HoA and CoA is correct (The both address are routed to the MN). This is verified by sending test signaling message from both of HoA and CoA (thus via HA and direct to CN). *Home Test Init (HoTI)* and *Home Test (HoT)* are the request and reply message via the HA respectively, and *Care-of Test Init (CoTI)* and *Care-of Test (CoT)* as well respectively. Once return routability suscesses, CN maintain Binding Cache and the MN store CN address in the Binding Update List. The MN sends Binding Update to both the HA and the CN.

25.1.3 NEMO

The NEMO Basic Support [2] is the network mobility support protocol specified at IETF, whicle Mobile IPv6 is host mobility. NEMO is designed based on Mobile IPv6. To support network mobility, a router called *Mobile Router (MR)* manage mobility in behalf of all the nodes in mobile network. Thus the nodes inside the mobile network named *Moble Network Nodes (MNN)* are standard IPv6 nodes without mobility management functionalities. The MR send Binding Update to HA like as Mobile IPv6, but in NEMO, *Mobile Network Prefix (MNP)* are included in the message. This prefix is stored in HA as NEMO prefix table with Binding Cache. The packets from CN that destinated to the Mobile Network Prefix are captured at HA and forwarded to the MR with IP-in-IP tunnel. The MR decapsulates the tunnel and send it to the MNN. The signaling messages between nodes and Binding database are illustrated in figure 30.

25.1.4 Multiple Care-of Address Registration

Mobile IPv6 and NEMO basic support configure a tunnel between HA address and CoA of MN and MR respectively, even if MN and MR has several network interfaces. This is because an HoA correspond a CoA in these mobility technologies. Multiple Care-of Addresses Registration (MCoA) [13] is thus proposed as an extension of both Mobile IPv6 and NEMO Basic Support to establish multiple tunnels between MR and HA. Each tunnel is distinguished by its *Binding Identification number (BID).* The muliple CoAs are registered with BID in the Binding Cache at HA and CN as in figure 30. In other words, Mobile IPv6 and NEMO Basic Support only realizes interface switching while MCoA supports simultaneous use of multiple interfaces. An MN and an MR can register multiple CoAs at once by sending a single BU to the HA, that defined as bulk registration. This is useful to save the number of the signaling messages between Nodes. The bulk registration is currently specified only for home registration.

25.2 Classification of Route Optimization in NEMO

There are several propositions for Route Optimization in NEMO context. They cover a broad range of topics in terms of scenarios, benefits and disadvantages. Thus this section describes comparison of multiple approaches. First scenario is described, and then the approaches are classified into five types. Last of all, these approaches are compared. Summary is shown at Table 6.

25.2.1 Taxonomy

In NEMO context, Route Optimization is divided into two scenarios that are **Non-Nested Scenario** and **Nested Scenario**. In Non-nested scenario, the issues that are similar in Mobile IPv6 tend to be focused, such as Longer Route, Packet Encapsulation and Bottleneck in the HA that are shown in Section 25.2. On the other hand, in Nested Scenario, the focus is on the issues, which do not appear in Mobile IPv6 but NEMO. This is because Nested Mobile Network further amplifies the issues listed above in Non-Nested Scenario.

In Non-nested scenario, there are two approaches from the way to optimize the route. Both approaches have common idea that the Binding Cache is transplanted to a router closer to Correspondent Nodes. "(a) Binding Management on CE" is the Binding Cache is transplanted to Correspondent Entities. On the other hand, in "(b) Infrastructure-based RO", it is transplanted to the nearer HA from the MR. In nested scenario, Some of the HAs are skipped by the solutions. One approach is to use topological Care-of Address that is classified to "(e) Topological CoA relay". The other approaches which aim nested scenario are classified to "(d) Halfway Home Agents Skip". "(c) RO using MANET" is only one approach to aim to both of the two scenarios.

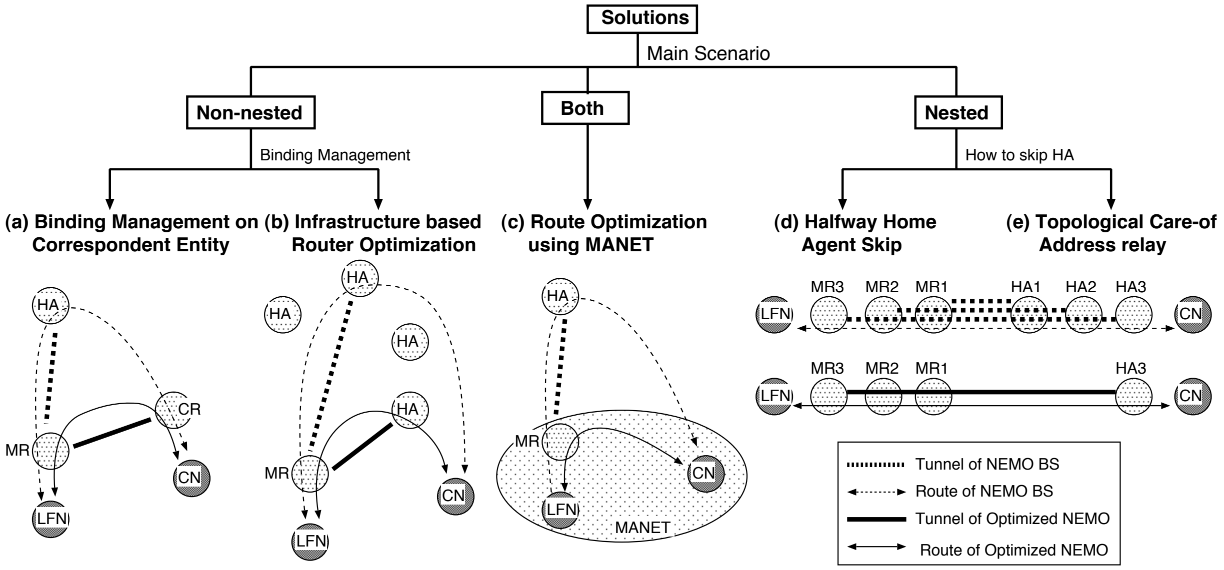


Figure 31: Approaches of Route Optimization

Table 6: Relation with target scenario and each approach

Main scenario	Approach	Examples	(1)	(2)	(3)	(4)
Non-nested	(a) Binding Management on CE	[14, 15, 16, 17](,[18][1])	○	×	○	×
	(b) Infrastructure-based RO	[19, 20]	○	×	○	×
Both	(c) RO using MANET	[21, 22, 23, 24]	○	○	○	○
Nested	(d) Halfway Home Agent Skip	[25, 26, 27]	×[2]	×[2]	×[2]	○
	(e) Topological CoA relay	[28, 29, 30]	×[2]	×[2]	×[2]	○

(1) Longer Route
(2) Packet Encapsulation
(3) Bottleneck in the Home Agent
(4) Nested Mobile Networks

25.2.2 Binding Management on Correspondent Entity

An orthodox approach to Route Optimization in NEMO is for the MR to attempt Route Optimization with a Correspondent Entity as (a) in Figure 31. The Correspondent Entity, having receivedgi the Binding Update, can then set up a bi-directional tunnel with the MR at the current Care-of Address of the MR. This approach is similar idea with Route Optimization in Mobile IPv6 that Binding Cache management functionality is transplanted from the HA to Correspondent Entity.

[14, 15, 16, 17] are examples of this approach. They mainly focus on Non-nested scenario, thus issue may still remain in Nested Scenario. [31] further investigated for the

---

[1] Support Nested case by assuming local routing among MRs

[2] Issues are alleviated by reduced tunnel overhead

approach. [18] focuses both of Non-nested and nested scenario by assuming existence of the local routing protocol within nested mobile networks. But main idea of this proposal must be classified to Binding Management on Correspondent Entity.

Since Correspondent Entity assumes to be closer to the Correspondent Node than the HA, Longer Route is optimized. And Bottleneck in the HA is solved, because the tunnel is created between the MR and the Correspondent Entity instead of the HA. On the other hand, Packet Encapsulation issue is still untouched.

25.2.3 Infrastructure-based Route Optimization

Infrastructure-based Route Optimization is a type of approach that transplants Binding Cache management functionality to a router close to the Correspondent Node instead of the initial HA as (b) in Figure 31. One example is to make use of Mobility Anchor Points (MAPs) such as defined in Hierarchical Mobile IPv6 [20]. Another example is to make use of proxy HA such as defined in the global Home Agent to Home Agent (HAHA) protocol [19].

Longer Route is optimized by Binding Cache being managed by closer router such as MAPs or proxy HAs. And Bottleneck in the HA is solved, because the tunnel is created between the MR and another router instead of the initial HA. On the other hand, the Packet Encapsulation issue is still untouched. Nested mobility optimization needs additional scheme to be solved.

25.2.4 Route Optimization using MANET

Route Optimization using MANET is the approach of local packet delivery in MANET instead of NEMO as (c) in Figure 31. In other words, this is the MANEMO case. The example is [21, 22, 23, 24]. This assume that both MRs and Correspondent Entity support both of NEMO and MANET technologies and they exchange direct route when they are connecting in same MANET cloud. Thus the path of communication is switched to direct route from non-optimized route, when destination and source of communication are in same MANET cloud.

By using direct route, all the HAs and tunnels are skipped in the both cases of Non-nested and Nested. Thus communication is free from Longer Route, Packet Encapsulation and Bottleneck in the HA in the both cases. However, this optimization can be utilized only in local MANET area. Detailed problem statement was described in [32, 33, 34] in MANEMO WG [35] in IETF.

25.2.5 Halfway Home Agent Skip

Halfway Home Agents Skip mainly focuses on nested scenario to reduce number of tunnels and number of HAs on the path as (d) in Figure 31. Examples are [25, 26] that the tunnel ends up between MR3 and HA3 by using Reverse Routing Header (RRH) in Figure 31. On the other hand, the tunnel is end up between MR1 and HA3 in [27]. Those three examples must be classified to Nested Mobility Optimization approach with slight different, because all of them has an idea to skip the some of HAs and the tunnels in Nested mobile network.

By skipping the HAs and tunnels, the performance of a nested mobile network is decreased to almost the same level as NEMO basic support. But some sub-optimality still exists at the same level as NEMO basic support such as Longer Route, Packet Encapsulation and Bottleneck in the HA.

25.2.6 Topological Care-of Address relay

Topological Care-of Address (CoA) relay is mainly focus nested scenario to reduce number of tunnels and number of HAs on the path as (e) in Figure 31. This approach is divided into two types that are with Prefix Delegation (PD) [36] and with Neighbor Discovery Protocol (NDP) [37] proxy. Former way is for parent MRs to have functionality of Prefix Delegation. Examples of this are [28, 29]. MRs in nested mobile network acquire its Care-of Address that is from an aggregatable address space starting from the access router by prefix delegation. Since the Care-of Address is routable without both of HA1-MR1 tunnel and HA2-MR2 tunnel, finally only tunnel between MR3 and HA3 is established.

Example of later way with NDP proxy is [30]. A MR relays the prefix of its care-of address to the nodes behind the MR. All MRs in nesting will configure a care-of address from the network prefix advertised by its access router. The entire mobile network and its access network form a logical multi-link subnet, thus eliminating any nesting. In both types in this approach, by skipping the HAs and tunnels, the performance of a nested mobile network is decreased to almost the same level as NEMO basic support. But the same sub-optimality still exists as NEMO basic support such as Longer Route, Packet Encapsulation and Bottleneck in the HA.

References

[1] R. Hinden and S. Deering. IP Version 6 Addressing Architecture. RFC 4291 (Draft Standard), February 2006.

[2] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert. Network Mobility (NEMO) Basic Support Protocol. RFC 3963 (Proposed Standard), January 2005.

[3] R. Coltun, D. Ferguson, J. Moy, and A. Lindem. OSPF for IPv6. RFC 5340 (Proposed Standard), July 2008.

[4] G. Malkin and R. Minnear. RIPng for IPv6. RFC 2080 (Proposed Standard), January 1997.

[5] T. Narten, E. Nordmark, W. Simpson, and H. Soliman. Neighbor Discovery for IP version 6 (IPv6). RFC 4861 (Draft Standard), September 2007.

[6] T. Clausen and P. Jacquet. Optimized Link State Routing Protocol (OLSR). RFC 3626 (Experimental), October 2003.

[7] R. Vida and L. Costa. Multicast Listener Discovery Version 2 (MLDv2) for IPv6. RFC 3810 (Proposed Standard), June 2004. Updated by RFC 4604.

[8] D. Johnson, C. Perkins, and J. Arkko. Mobility Support in IPv6. RFC 3775 (Proposed Standard), June 2004.

[9] C. Ng, P. Thubert, M. Watari, and F. Zhao. Network Mobility Route Optimization Problem Statement. RFC 4888 (Informational), July 2007.

[10] C. Ng, F. Zhao, M. Watari, and P. Thubert. Network Mobility Route Optimization Solution Space Analysis. RFC 4889 (Informational), July 2007.

[11] R. Baldessari, T. Ernst, A. Festag, and M. Lenardi. *Automotive Industry Requirements for NEMO Route Optimization.* IETF, January 2009. Work in progress, draft-ietf-mext-nemo-ro-automotive-req-02.

[12] W. Eddy, W. Ivancic, and T. Davis. *NEMO Route Optimization Requirements for Operational Use in Aeronautics and Space Exploration Mobile Networks*, December 2007. IETF, draft-ietf-mext-aero-reqs-00.

[13] R. Wakikawa, T. Ernst, K. Nagami, and V. Devarapalli. *Multiple Care-of Addresses Registration*, January 2008. IETF, draft-ietf-monami6-multiplecoa-05.

[14] R. Wakikawa, S. Koshiba, K. Uehara, and J. Murai. Orc: optimized route cache management protocol for network mobility. In *Proc. 10th International Conference on Telecommunications ICT 2003*, volume 2, pages 1194–1200, 23 Feb.–1 March 2003.

[15] J. Na, S. Cho, C. Kim, S. Lee, H. Kand, and C. Koo. *Route Optimization Scheme based on Path Control Header*, April 2004. IETF, draft-na-nemo-path-control-header-00.

[16] Carlos J. Bernardos, Maria Calderon, and Ignacio Soto. *Correspondent Router based Route Optimisation for NEMO (CRON)*, July 2008.

[17] Carlos J. Bernardos, Maria Calderon, and Ignacio Soto. *Mobile IPv6 Route Optimisation for Network Mobility (MIRON)*, July 2008.

[18] M. Watari, T. Ernst, and R. Wakikawa. *Routing Optimization for Nested Mobile Networks*, October 2006. IEICE Trans. Commun. (IEICE) Volume E89-B, Issue 10.

[19] P. Thubert, R. Wakikawa, and V. Devarapalli. *Global HA to HA protocol*, September 2006. IETF, draft-thubert-nemo-global-haha-02.

[20] H. Soliman, C. Castelluccia, K. El Malki, and L. Bellier. Hierarchical Mobile IPv6 Mobility Management (HMIPv6). RFC 4140 (Experimental), August 2005. Obsoleted by RFC 5380.

[21] M. Tsukada and T. Ernst. *Vehicle Communication Experiment Environment With MANET And NEMO*, January 2007. SAINT WONEMO 2007, Hiroshima, Japan.

[22] R. Wakikawa, K. Okada, R. Koodli, A. Nilsson, and J. Murai. Design of Vehicle Network: Mobile Gateway for MANET and NEMO Converged Communication, September 2005. The Second ACM International Workshop on Vehicular Ad Hoc Networks (VANET 2005).

[23] J. Lorchat and K. Uehara. Optimized Inter-Vehicle Communications Using NEMO and MANET (Invited Paper), July 2006. The Second International Workshop on Vehicle-to-Vehicle Communications 2006 (V2VCOM 2006).

[24] Carlos Jesus Bernardos, Ignacio Soto, María Calderón, Fernando Boavida, and Arturo Azcorra. Varon: Vehicular ad hoc route optimisation for nemo. *Computer Communications*, 30(8):1765–1784, 2007.

[25] P. Thubert and M. Molteni. *IPv6 Reverse Routing Header and its application to Mobile Networks*, February 2007. IETF, draft-thubert-nemo-reverse-routing-header-07.

[26] J. Na, S. Cho, C. Kim, S. Lee, H. Kand, and C. Koo. *Secure Nested Tunnels Optimization using Nested Path Information*, September 2003. IETF, draft-na-nemo-nested-path-info-00.

[27] C. Ng and J. Hirano. *Securing Nested Tunnels Optimization with Access Router Option*, July 2004. IETF, draft-ng-nemo-access-router-option-01.

[28] E. Perera, R. Hsieh, and A. Seneviratne. *Extended Network Mobility Support*, July 2003. IETF, draft-perera-nemo-extended-00.

[29] K. Lee, J. Jeong, J. Park, and H. Kim. *Route Optimization for Mobile Nodes in Mobile Network based on Prefix Delegation*, February 2004. IETF, draft-leekj-nemo-ro-pd-02.

[30] J. Jeong, K. Lee, and H. Kim. *ND-Proxy based Route and DNS Optimizations for Mobile Nodes in Mobile Network*, February 2004. IETF, draft-jeong-nemo-ro-ndproxy-02.

[31] R. Wakikawa. *The Design Consideration of Correspondent Router*. IETF, July 2008. Work in progress, draft-wakikawa-mext-cr-consideration-00.

[32] R. Wakikawa, P. Thubert, T. Boot, J. Bound, and B. McCarthy. Problem statement and requirements for manemo, 2007.

[33] R. Wakikawa, T. Clausen, B. McCarthy, and A. Petrescu. Manemo topology and addressing architecture, 2007.

[34] B. McCarthy, A. Petrescu, and T. H. Clausen. *MANEMO Configuration Problems (MCP)*, March 2007. IETF, draft-mccarthy-manemo-configuration-problems-00.

[35] IETF: Mobile Ad-hoc Networks (MANET) Working Group, 2008. http://www.ietf.org/html.charters/manet-charter.html.

[36] O. Troan and R. Droms. IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6. RFC 3633 (Proposed Standard), December 2003.

[37] T. Narten, E. Nordmark, and W. Simpson. Neighbor Discovery for IP Version 6 (IPv6). RFC 2461 (Draft Standard), December 1998. Obsoleted by RFC 4861, updated by RFC 4311.