PAPER

An Identifier Locator Separation Protocol for the Shared Prefix Model over IEEE WAVE IPv6 Networks

Sangjin NAM[†], *Nonmember and* Sung-Gi MIN[†a)], *Member*

SUMMARY As the active safety of vehicles has become essential, vehicular communication has been gaining attention. The IETF IPWAVE working group has proposed the shared prefix model-based vehicular link model. In the shared prefix model, a prefix is shared among RSUs to prevent changes in IPv6 addresses of a vehicle within a shared prefix domain. However, vehicle movement must be tracked to deliver packets to the serving RSU of the vehicle within a shared prefix domain. The Identifier/Locator Separation Protocol (ILSP) is one of the techniques used to handle vehicle movement. It has several drawbacks such as the inability to communicate with a standard IPv6 module without special components and the requirement to pass signaling messages between end hosts. Such drawbacks severely limit the service availability for a vehicle in the Internet. We propose an ILSP for a shared prefix model over IEEE WAVE IPv6 networks. The proposed protocol supports IPv6 communication between a standard IPv6 node in the Internet and a vehicle supporting the proposed protocol. In addition, the protocol hides vehicle movement within a shared prefix domain to peer hosts, eliminating the signaling between end hosts. The proposed protocol introduces a special NDP module based on IETF IPWAVE vehicular NDP to support vehicular mobility management within a shared prefix domain and minimize link-level multicast in WAVE networks.

key words: *shared prefix model, WAVE, identifier/locator separation protocol, IPWAVE, V2I*

1. Introduction

Interest in vehicular communication to support the active safety of vehicles has been increasing. By using it, vehicles may provide their state information to other vehicles or gather traffic information from infrastructure such as traffic light. This information can be used to mitigate vehicle damage caused by traffic accidents. However, the location of a vehicle changes dynamically. This may cause changes in IPv6 addresses of a vehicle, leading to disruption of the IP connectivity of the vehicle to an IPv6 node in the Internet.

To mitigate this problem, the IETF IP Wireless Access in Vehicular Environments Working Group (IPWAVE WG) has proposed the shared prefix model [1] considering the need for minimizing link-level multicast [2] and wireless link properties such as the asymmetric link property and the undetermined link-level connectivity. In this model, multiple roadside units (RSUs) cover a shared prefix domain, where the RSUs use the same prefix for vehicles in their coverage. Within a shared prefix domain, global IPv6 addresses of vehicles do not change despite their movement. Because IPv6 addresses do not change, the IP connectivity between vehicles and their corresponding nodes (CNs) is not disrupted. However, the data path within the shared prefix domain changes; therefore, vehicle movement must be tracked to deliver packets to vehicles correctly.

One of the major techniques for handling the movement of a mobile node (MN) in the Internet is the Identifier/Locator Separation Protocol (ILSP), which includes the Host Identity Protocol (HIP) [3], Locator/ID Separation Protocol (LISP) [4] and Identifer/Locator Network Protocol (ILNP) [5]. It separates the identifier and locator. The identifier distinguishes an MN itself. The locator indicates the current location of the MN. ILSP supports the mobility of an MN by using only the identifier as a connection endpoint.

However, there are major drawbacks to the well-known ILSP variants. First, they cannot communicate with standard IPv6 nodes because of the different semantics of the endpoints in each of these protocols. Secondly, they require a special or additional mapping server to resolve the current locator of an MN from its identifier or a Fully Qualified Domain Name (FQDN). Even if they use the standard DNS [6], they introduce special resource records (RRs) [7]. They also require special signaling messages between peer nodes to track changes in the locator of an MN [8]–[10].

The IPWAVE WG investigated the well-known ILSP variants for application to IPWAVE networks [11]. It mentions that mobility management, security and privacy must be considered for ILSP over vehicular networks.

We propose an ILSP for the shared prefix model over IEEE WAVE IPv6 networks. The proposed scheme adapts the shared prefix model [1] to IEEE WAVE IPv6 networks. Because IPv6 addresses of a vehicle do not change within a shared prefix domain, the IP connectivity between a CN and vehicle is not disrupted until the vehicle leaves the shared prefix domain. Consequently, special signaling messages between the CN and vehicle are not required to track the current location of the vehicle.

The protocol supports host-initiated vehicular mobility management (VMM) to handle data path changes due to vehicle movement within a shared prefix domain. It exploits the ILNPv6 address structure. The global IPv6 address of a vehicle consists of the 64-bit locator and identifier. Within a shared prefix domain, the protocol uses the locator part of the destination IPv6 address of a packet to route the packet to the serving RSU of the vehicle, without an IPv6 encapsulation mechanism. The protocol hides the usage of the locator to the vehicle and CN.

Our protocol introduces a special Neighbor Discovery

Manuscript received August 24, 2022.
Manuscript publicized October 21, 2022.

[†]The authors are with Department of Computer Science and Engineering, Korea University, Seoul, South Korea.
a) E-mail: sgmin@korea.ac.kr (Corresponding author)

DOI: 10.1587/transcom.2022EBP3141

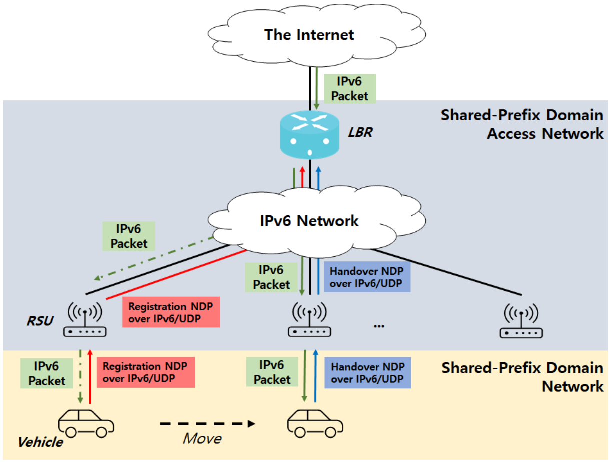

The Internet
IPv6 Packet
Shared-Prefix Domain Access Network
LBR
IPv6 Network
IPv6 Packet
IPv6 Packet
Handover NDP over IPv6/UDP
RSU
Registration NDP over IPv6/UDP
...
IPv6 Packet
Registration NDP over IPv6/UDP
IPv6 Packet
Handover NDP over IPv6/UDP
Shared-Prefix Domain Network
Vehicle
Move

Fig. 1 The overall structure of our VMM scheme.

Protocol (NDP) module based on [2]. Vehicles, RSUs, and the link border router (LBR) support the special NDP module, which is used to register IPv6 addresses of a vehicle, perform Duplicate Address Detection (DAD), and manage vehicular mobility within a shared prefix domain. The special NDP uses UDP to separate itself from the standard NDP [12], which uses ICMPv6. It also combines multiple special NDP messages into a single UDP message for the simultaneous registration and/or handover notification of multiple IPv6 addresses. Figure 1 shows the overall structure of our VMM scheme. Using special NDP messages, vehicles notify their current serving RSU to the LBR. The LBR forwards a packet to the serving RSU notified by the messages. In addition, all vehicles in the shared-prefix domain WAVE network support IPv6 using V2I communication with RSUs.

The proposed protocol uses a human-readable identity, which conforms to the Network Access Identifier (NAI) [13], for a vehicle. The protocol supports mapping from the human-readable identity to the identifier and current locator (an IPv6 address) using the standard DNS service discovery (DNS-SD) [14] and DNS AAAA query/response without any additional extension.

The proposed protocol does not use the modified EUI-64 address for the interface identifier (IID) of an IPv6 address. Our protocol uses the stable and opaque IID generation method [15] recommended in [16]. Therefore, an IPv6 address is not affected by changes in a link-layer address and a link-layer address cannot be inferred from an IPv6 address.

The remainder of this paper is organized as follows. Section 2 discusses WAVE specified in IEEE Std 1609, vehicular networks proposed by the IPWAVE WG, the well-known ILSP variants, and considerations in the WAVE link model. Section 3 describes the overall design of the proposed protocol. Section 4 describes how the proposed protocol operates in detail. Section 4 presents the simulation results of the protocol implemented in the Network Simulator 3 (NS-3) [17]. Finally, Section VI concludes the study and discusses future work.

2. Related Works

2.1 IEEE WAVE Standards and IETF IPWAVE Working Group

IEEE Std 1609 defines a family of standards for WAVE. It enables secure vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) wireless communications. IEEE 1609.3 [18] defines networking services for WAVE. The WAVE standard supports two network/transport layer protocol stacks: IPv6-based protocols and the WAVE Short Message Protocol (WSMP). For V2I communication, an RSU acts as a default gateway for on-board units (OBUs) in a vehicle located in the coverage area of the RSU. An RSU periodically broadcasts a WAVE Service Advertisement (WSA). If a WSA contains the IPv6 routing service, it also contains a WAVE Router Advertisement (WRA). The WRA presents IPv6 configuration information such as a prefix, default gateway IPv6 and link-layer addresses. Using this information, a WRA can replace the standard Router Advertisement (RA) and link-layer address resolution of the default gateway IPv6 address. An OBU chooses a WSA which contains the IPv6 routing service and configures its IPv6 module using the WRA in the WSA.

Annex D of IEEE 1609.0 [19] describes that collision avoidance applications periodically exchange SAE J2735 messages [20] such as Basic Safety Message (BSM) using WSMP. Since a BSM contains speed, heading and 3D location information of a vehicle and is expected to be sent via pre-determined channels including the control channel (CCH), vehicles can detect and warn an approaching vehicle. Similarly, these applications may be implemented over IPv6/UDP with pre-determined service channels (SCHs) and UDP port numbers since IPv6 traffic is not allowed on the CCH.

The IETF IPWAVE WG has developed IPv6-based solutions for V2V and V2I. The IPWAVE WG describes a problem statement for adopting current IPv6 protocols, such as the IPv6 NDP and mobility management protocol, to vehicular networks. It proposed a vehicular link model in the IPWAVE network [2] that uses the shared prefix model and handles wireless link properties such as the undetermined link-level connectivity. The IPWAVE WG also describes problems that occur when the well-known ILSP variants are applied to vehicular networks [11]. For the VMM [1], it proposes the mobility management scheme based on [21].

2.2 Identifier/Locator Separation Protocols

ILSP variants [3]–[5] are based on a paradigm which renders each OSI layer independent of IP address. They separate the identifier and locator parts from the current IP address architecture.

HIP [3] uses a public certificate as an identity and its hashed tag (HIT) is bound to network and transport layers. Because an HIT is not globally routable in the Internet, the locator for the destination HIT is needed to send messages to the destination. HIP introduces a rendezvous server (RVS) [22] to track HIP hosts in a specific domain. Each HIP host discovers the RVS using a DNS query with the FQDN of the destination HIP host and sends an initial HIP message, included as the IPv6 extension header, to its RVS and the RVS forwards the message to the destination if the destination HIT is registered at the RVS. HIP requires that end hosts must be HIP-capable and mobile hosts must register their current locators to its RVS. In addition, a data packet is transported in the ESP transport mode and the network layer module of an HIP node must translate between the HIT and the IPv6 address in the packet before the upper-layer processing.

LISP [4] defines the new address space, called the Endpoint Identifier (EID) address space. LISP hosts have their own EIDs and use them as identifiers. An EID is bound to network and transport layers. An EID is not globally routable in the Internet but is locally routable within its LISP site. There is a tunnel router at the boundary of each LISP site. If an LISP host sends a packet to another LISP host within a different LISP site, the packet first arrives at the ingress tunnel router (ITR) of the source LISP host. The ITR finds the routing locator (RLOC) via the MAP server to forward the packet to the egress tunnel router (ETR) which manages the LISP site where the destination LISP host belongs. Subsequently, the ITR encapsulates the packet with the RLOC and sends it to the ETR. The ETR decapsulates the packet and locally forwards it to the destination LISP host using the destination EID. By this procedure, however, a non-LISP host cannot communicate with an LISP host if special routers, called proxy ITR/ETR [23], are not used. Moreover, to cope with LISP host movement, LISP requires an extra mapping server to map the EID and its current locator. An LISP mobile host must register its EID and locator to its mapping server and decapsulate packets tunneled to itself [9].

ILNPv6 [5] exploits the standard IPv6 address structure, as shown in Fig. 2. If the location of a node changes, its 64-bit locator part of the address is changed with the network prefix of the point of attachment. The change is notified to CNs by locator update messages [10]. To avoid connection disruption by locator changes, an ILNPv6 connection endpoint only includes the identifier part. As a result, the semantics of a connection endpoint in standard TCP/IP protocols are redefined to include only the identifier part of an IPv6 address in the pseudo header. Additionally, an ILNPv6 node uses the DNS server with the FQDN of the destination ILNPv6 node to resolve the current locator and identifier. For this purpose, new RRs have been introduced [7]. An ILNPv6 node must register its locator and identifier to the DNS server in its domain using its FQDN, and send DNS update messages [24] to the DNS server whenever its locator changes.

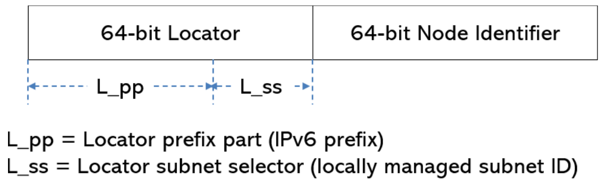

64-bit Locator
64-bit Node Identifier
L_pp
L_ss
L_pp = Locator prefix part (IPv6 prefix)
L_ss = Locator subnet selector (locally managed subnet ID)

Fig. 2 The ILNPv6 address structure.

2.3 Considerations for the WAVE Link Model

The WAVE link model described in [25] exhibits asymmetric property. However, standard IPv6 NDP assumes that it operates only on symmetric links. In addition, [25] refers to [26], which defines an IP addressing model for ad-hoc networks and the undetermined link-level connectivity property. Considering them, [25] mentions that some considerations should be applied to the WAVE link model, as follows:

- An IP address configured on an interface should be unique, at least within the routing domain.
- No on-link subnet prefix should be configured on an interface.

As a result, a vehicle does not add a received prefix into the prefix list on its IPv6 module, rendering the prefix list empty. The on-link determination for any outgoing packet becomes off-link. Therefore, the next hop of any outgoing IPv6 packet is the default gateway.

3. Identifier Locator Separation Protocol for Shared Prefix Model over IEEE WAVE IPv6 Networks

Our proposed architecture is based on the shared prefix model proposed in [1]. Figure 3 shows the network topology of the proposed architecture. The address structure uses the 64-bit locator and identifier, based on the ILNPv6 address structure. The 64-bit locator consists of a 48-bit global routing prefix and 16-bit subnet identifier. As an ILNPv6 address is compatible with a standard IPv6 address, a standard IPv6 CN can use the ILNPv6 address as the peer IPv6 address for a vehicle.

A virtual link comprises multiple sub-links and is managed by the LBR, and each RSU manages its own sub-link. A virtual link has a shared prefix containing a global routing prefix and zero subnet identifier. The prefix of a virtual link is shared as an IPv6 prefix for all vehicles within the virtual link coverage. Each sub-link has a prefix which consists of the same global routing prefix and a nonzero subnet identifier. The global routing prefix is used as the global locator of a shared prefix domain in the Internet. The LBR advertises the global routing prefix to the Internet; consequently, all packets destined for the global routing prefix are first forwarded to the LBR. The subnet identifier acts as the local locator of a vehicle on the virtual link coverage. It is locally modified within the WAVE Access Network to indicate the current serving RSU of a destination vehicle. After a packet is arrived at the serving RSU, the modified subnet identifier is restored to the original value, zero. As a result, from the viewpoint of a vehicle and the Internet, it seems that there

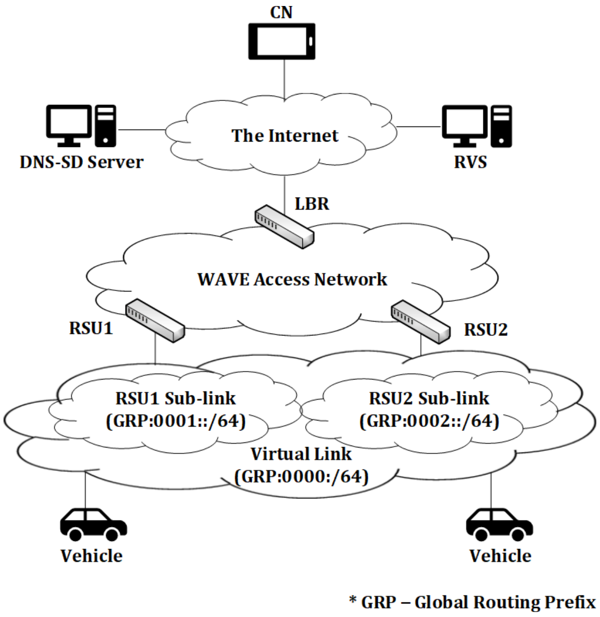


Fig. 3 The proposed network topology.

is no modification to packets. Therefore, unlike a traditional Network Address Translator (NAT) [27], our scheme may use security protocols which guarantee integrity of an IPv6 header between a vehicle and a CN. It does not need any traversal utility such as the Session Traversal Utilities for NAT (STUN) [28]. In addition, any IPv6 tunneling mechanism, which can overload the LBR [29], is not used for data packet forwarding between the LBR and an RSU.

A vehicle generates global IPv6 addresses based on the shared prefix. To generate an IPv6 address, the vehicle generates an IID using the stable and opaque IID generation method [15] recommended by [16]. Subsequently, it generates an IPv6 address by concatenating the shared prefix and generated IID. The generated IPv6 address is independent of the link-layer address of the vehicle, because the IID is independent of the link-layer address.

A generated IPv6 address is registered using our special NDP module. Successful registration means that the LBR which manages the virtual link has confirmed no duplication of the IPv6 address and the IPv6 address becomes the valid (preferred) address. The vehicle may use the IPv6 address as its current IPv6 address; therefore, it updates the IPv6 address to its RVS.

Whenever a vehicle handovers to a new RSU within the same virtual link, it updates its current serving RSU to the LBR using the special NDP module.

Our special NDP module is based on [2]. Because of the relatively narrow bandwidth and wider coverage of WAVE channels than that of standard WiFi channels, excessive traffic should be constrained. Therefore, the special NDP module uses only unicast messages at the link-layer even though destination IPv6 addresses of the messages are multicast IPv6 addresses [30]. It also uses UDP and combines several address registration and/or handover messages into a single UDP message. Consequently, the special NDP can coexist with the standard NDP which uses ICMPv6.

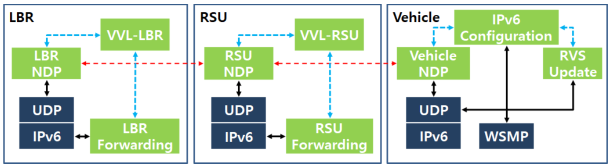


Fig. 4 The proposed modules in our protocol.

Following subsections explain each component in Fig. 3. Each component has own data structure and modules to support the proposed architecture.

3.1 Vehicle

A vehicle is an MN which may want to access a service on the Internet. We assume that each vehicle has an NAI of which the domain part conforms to the standard DNS [6]; therefore, the NAI can be converted to the form of the domain name. In following sections, we call the converted form of the NAI as the converted NAI.

The modules of a vehicle are shown in Fig. 4. The IPv6 configuration module configures its IPv6 stack and continuously monitors WSAs. It follows the stateless address autoconfiguration (SLAAC) [31] and uses the recommended stable IID generation method [15], [16]. The IPv6 configuration module also performs address registration by calling the vehicle NDP module. The vehicle NDP module registers link-local and global IPv6 addresses to the LBR via the serving RSU of the vehicle. If address registration is successful, the RVS update module is called to update the registered global IPv6 address to the RVS.

When the IPv6 configuration module receives a new WSA from another RSU within the same virtual link, it performs a handover decision procedure. If it decides to handover to the new RSU, it calls the vehicle NDP module to notify the handover to the LBR via the new RSU.

3.2 Road-Side Unit (RSU)

An RSU acts as a default gateway for vehicles in its coverage. It contains RSU-specific modules illustrated in Fig. 4.

The visiting vehicle list in the RSU (VVL-RSU) module maintains the special cache, called VVL-RSU. Each VVL-RSU entry consists of the IPv6 and link-layer address of a vehicle and is added or deleted by the RSU NDP module. The VVL-RSU acts as the neighbor cache of the WAVE interface of an RSU.

The RSU NDP module performs address registration procedure on behalf of a vehicle. If address registration is successful, the RSU NDP module stores a pair of <a new IPv6 address, the corresponding link-layer address> in the VVL-RSU.

The RSU forwarding module forwards a data packet to a destination vehicle located in the coverage of the RSU using the VVL-RSU module.

3.3 Link Border Router (LBR)

The LBR connects the virtual link to the Internet. LBR-specific modules are illustrated in Fig. 4.

The special cache called visiting vehicle list in the LBR (VVL-LBR) is maintained by the VVL-LBR module. It stores information of all vehicles within its virtual link. Each VVL-LBR entry consists of the IPv6 address of a vehicle and the subnet identifier of the RSU serving the vehicle. VVL-LBR entries are added or deleted by the LBR NDP module.

The LBR NDP module performs address registration procedure, which replaces the DAD of registered IPv6 addresses within the virtual link. If the registering IPv6 address is not already registered, it stores a pair of <a new IPv6 address, the subnet identifier of the serving RSU> in the VVL-LBR.

The LBR forwarding module forwards a data packet, destined for a vehicle within the virtual link, to the current serving RSU of the vehicle using the VVL-LBR module.

3.4 Rendezvous Server (RVS)

An RVS is a DNS server for a domain to which a vehicle belongs. It maps the converted NAI of a vehicle to its current IPv6 address. The standard DNS query/response is used for the mapping service.

3.5 Corresponding Node (CN)

A CN is a host that communicates with a vehicle in a WAVE network. It may be a host in the Internet or another vehicle. A CN is assumed to know the NAI of the vehicle which it wants to communicate with.

In our architecture, a CN requires a special resolver (RVS-RESOLVER). The resolver discovers the RVS for a vehicle using the DNS-SD procedure with the domain part of the NAI of the vehicle. Subsequently, it sends a DNS AAAA query for the converted NAI of the vehicle to the RVS.

4. Mechanisms in the Proposed Protocol

The proposed protocol consists of Configuration, RVS update, and Communication phases, as illustrated in Fig. 5.

4.1 Configuration Phase

4.1.1 Entering Into a New Virtual Link

When a vehicle enters a new virtual link, the IPv6 configuration module in the vehicle collects WSAs sent by surrounding RSUs. Among them, it selects the WSA which contains the IPv6 routing service and WRA. The selection may use the 3D-location information and/or signal strength. The IPv6 configuration module adds new Destination Cache Entry (DCE) and Neighbor Cache Entry (NCE) for the RSU by using the WRA in the selected WSA. As a result, the standard NDP for the link-layer address resolution of the RSU IPv6 address is not needed.

4.1.2 Generating IPv6 Addresses

The IPv6 configuration module must configure at least two IPv6 addresses: a link-local IPv6 address and global IPv6 address based on the shared prefix obtained from the WRA in the selected WSA. First, it generates a stable IID for each IPv6 address using the stable and opaque IID generation method [15]. For the link-local IPv6 address, the shared prefix is used as the Network_ID parameter of the pseudo-random function (PRF), because the link-local IPv6 address is for the virtual link. After the generation of stable IIDs, the IPv6 configuration module generates link-local and global IPv6 addresses.

4.1.3 Sending an UDP-NS at the Vehicle

The generated IPv6 addresses are tentative and cannot be used as valid IP addresses. Therefore, DAD must be performed for these addresses. The proposed scheme uses address registration instead of the DAD. The IPv6 configuration module requests address registration for newly generated IPv6 addresses to the vehicle NDP module. The vehicle NDP module generates a Neighbor Solicitation (NS) message for each IPv6 address. An NS message includes the registration code value, a registering IPv6 address in the target IPv6 address and corresponding link-layer address in the Source Link-Layer Address Option (SLLAO). It then combines these NS messages into one UDP message (UDP-NS) and sends the UDP-NS message to the RSU NDP module of the selected RSU. As there is no valid IPv6 address for the outgoing WAVE interface, the unspecified IPv6 address is used as the source IPv6 address of the UDP-NS message.

4.1.4 Processing the UDP-NS at the RSU

When the RSU NDP module on the RSU receives the UDP-NS message, it checks whether a VVL-RSU entry for the target IPv6 address at each NS message in the UDP-NS message exists. If so, it generates a Neighbor Advertisement (NA) message with an error code duplicated. The contents of the NA message, except for Type and Code, are the same as those of the corresponding NS message. Otherwise, it stores an NS message. If there are NA messages for address duplication notification, it combines them into one UDP message (UDP-NA) and sends it to the vehicle NDP module. The destination IPv6 address of the UDP-NA uses the all-node multicast IPv6 address; however, it uses the link-layer address included in the SLLAO in any NS message in the UDP-NS message. Such address mapping of an IPv6 multicast address to a link-layer unicast address was permitted by [30]. If there are stored NS messages, it combines them into one UDP-NS message again and sends

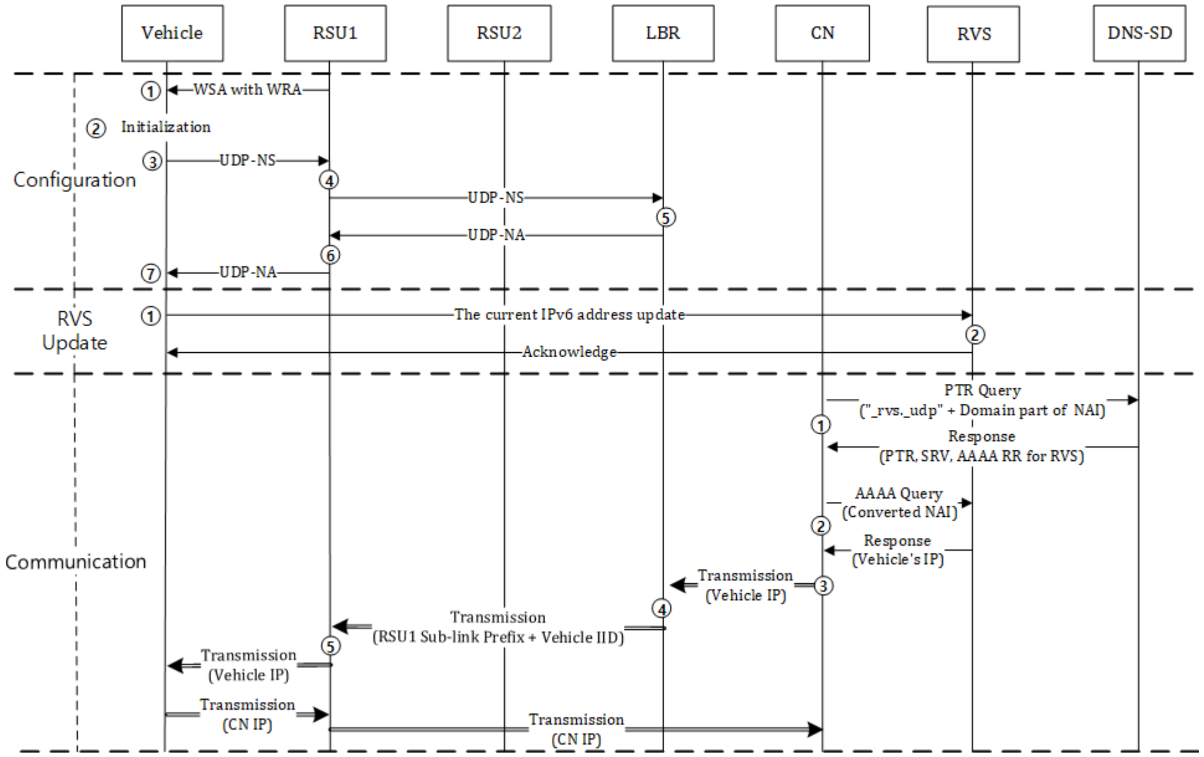

Vehicle
RSU1
RSU2
LBR
CN
RVS
DNS-SD
Configuration
① WSA with WRA
② Initialization
③ UDP-NS
④ UDP-NS
⑤ UDP-NA
⑥ UDP-NA
⑦
RVS Update
① The current IPv6 address update
② Acknowledge
Communication
PTR Query ("_rvs._udp" + Domain part of NAI)
①
Response (PTR, SRV, AAAA RR for RVS)
AAAA Query (Converted NAI)
②
Response (Vehicle's IP)
Transmission (Vehicle IP)
③
④
Transmission (RSU1 Sub-link Prefix + Vehicle IID)
⑤
Transmission (Vehicle IP)
Transmission (CN IP)
Transmission (CN IP)

Fig. 5 Communication phases in the proposed protocol.

it to the LBR NDP module. The source IPv6 address of the UDP-NS must be the IPv6 address of the WAVE interface through which the UDP-NS message is received. The prefix of the source IPv6 address must be the sub-link prefix of the RSU, which contains its 16-bit nonzero subnet identifier.

4.1.5 Processing the UDP-NS at the LBR

When the LBR NDP module receives the UDP-NS message, it looks up VVL-LBR entries for the target IPv6 address at each NS message in the UDP-NS message. If there is a matching entry, the LBR NDP module generates an NA message with an error code duplicated. Otherwise, it creates a new entry in the VVL-LBR with the target IPv6 address in the NS message and nonzero subnet identifier in the source IPv6 address of the message. Then it generates an NA message with a code success. The contents of the NA message, except for Type and Code, are the same as those of the NS message. It combines all NA messages into one UDP-NA message, and returns it to the RSU NDP module which sent the UDP-NS.

4.1.6 Processing the UDP-NA at the RSU

When the RSU NDP module receives the UDP-NA message, it processes each NA message in the UDP-NA message. If the code in the NA message is success, it creates a VVL-RSU entry with the target IPv6 address and link-layer address in the NA message. Then it forwards the UDP-NA, received from the LBR NDP module, to the vehicle NDP module using the all-node multicast IPv6 address with a link-layer address in any NA message in the UDP-NA.

4.1.7 Processing the UDP-NA at the Vehicle

When the UDP-NA arrives at the vehicle NDP module, the module checks the code of each NA message in the UDP-NA. Subsequently, using the code value, the results of the address registration are reported to the IPv6 configuration module. If the error duplicated is reported for a registering IPv6 address, the IPv6 configuration module regenerates a new stable IID by increasing DAD_Counter of the PRF, and repeats the address registration procedure. Otherwise, the registered IPv6 address becomes a preferred (or valid) address. It configures registered IPv6 addresses to its WAVE interface and sets its default gateway as the RSU.

4.2 RVS Update Phase

4.2.1 Sending a DNS Update to the RVS

After the IPv6 configuration module successfully configures its global IPv6 address, it calls the RVS update module to update its current IPv6 address to the RVS. The IPv6 address of the RVS can be discovered using the DNS-SD procedure with the NAI of the vehicle. The procedure is explained in detail in the following section. The DNS-SD server IPv6 address is included in the primary DNS field of the WRA. In addition, the RVS IPv6 address may be pre-configured. The DNS update message contains a pair of <the converted

NAI of the vehicle, the configured global IPv6 address>.

4.2.2 Receiving the DNS Update at the RVS

When the DNS update message arrives at the RVS, the RVS updates the AAAA RR for the vehicle according to the DNS update standard.

4.3 Communication Phase

4.3.1 Getting the IPv6 Address of the RVS for a Vehicle

An application on a CN interacts with the RVS-RESOLVER to get the current IPv6 address of a vehicle. The RVS-RESOLVER first constructs the RVS service name for the domain of the vehicle by concatenating _rvs._udp to the domain part of the NAI of the vehicle. Subsequently, it sends a DNS PTR query with the constructed RVS service name to its DNS-SD server. The answer section of a DNS response contains PTR RRs for RVS service instance names of the RVS service name. In the additional section, the AAAA RR and SRV RR of each RVS service instance name are included. An SRV RR and AAAA RR contain the service port number and IPv6 address of an RVS, respectively.

4.3.2 Getting the Current IPv6 Address of the Vehicle

Using the port number and IPv6 address of the RVS, the RVS-RESOLVER sends a DNS AAAA query for the converted NAI to the RVS. Then, the RVS responds with the DNS response containing the current IPv6 address of the vehicle.

4.3.3 Sending Packets from the CN to the Vehicle

The application in the CN sends a packet to the vehicle using the IPv6 address discovered in the previous step.

4.3.4 Processing the Packet at the LBR

When the LBR forwarding module receives a packet, it checks whether the packet is destined for a vehicle within its virtual link by comparing the prefix of the destination IPv6 address of the packet with its shared prefix. Note that the LBR advertises its global routing prefix, which is a superset of the shared prefix, to the Internet. If not, the packet is processed as an ordinary IPv6 packet. Otherwise, it checks whether there is a matching entry in the VVL-LBR for the destination IPv6 address of the packet. If a matching entry is not found, the packet is discarded. Otherwise, the LBR Forwarding module replaces the zero subnet identifier in the destination IPv6 address of the packet with the nonzero subnet identifier field of the matching entry. Then, it forwards the packet towards the WAVE access network.

4.3.5 Processing the Packet at the Serving RSU

When the RSU forwarding module receives a packet, it checks whether the packet is destined for a vehicle within its sub-link by comparing the prefix of the destination IPv6 address of the packet with its sub-link prefix. If not, it processes it as an ordinary IPv6 packet. Note that by performing this checking, packets from the vehicle to the CN are processed as ordinary IPv6 packets, and it does not cause the ingress filtering problem as source IPv6 addresses of packets belong to the virtual link. Otherwise, it checks whether the packet is destined for the interfaces of the RSU which have its sub-link prefix. If so, it processes the packet as an ordinary IPv6 packet. If not, it recovers the nonzero subnet identifier in the destination IPv6 address of each packets to zero. It then checks whether the vehicle is attached to the RSU by finding the VVL-RSU entry with the recovered destination IPv6 address. If so, it forwards the packet to the destination vehicle using the link-layer address in the matching VVL-RSU entry. Otherwise, the packet is discarded.

4.4 Intra-Link Mobility Management within the Virtual Link

Figure 6 shows the inter-sub-link mobility management procedure.

4.4.1 Entering the Coverage of Another RSU within the Same Virtual Link

If a vehicle moves into the coverage area of another RSU within the same virtual link, the IPv6 configuration module detects it by receiving WSAs which contains a WRA advertising the same shared prefix and a different default gateway information. It adds new DCE and NCE for the new RSU to avoid performing the standard ND procedure.

4.4.2 Sending an UDP-NS for Handover to the New RSU

If the IPv6 configuration module decides to handover to the

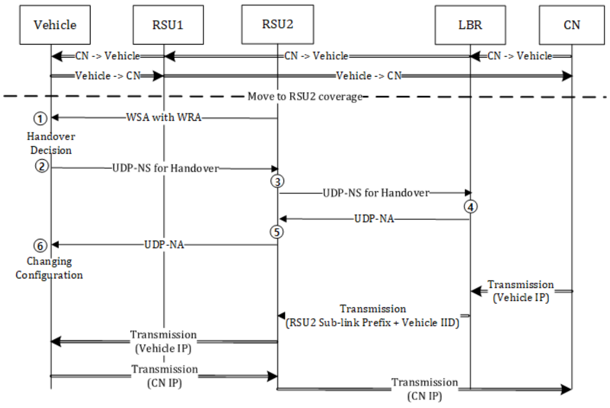


Fig. 6 Inter-sub-link handover.

new RSU, it calls the vehicle NDP module to notify handover to the new RSU. The vehicle NDP module generates NS messages, each of which contains the ‘handover’ code and a valid IPv6 address. The NS messages are combined into one UDP-NS message and the UDP-NS message is sent to the RSU NDP module of the new RSU. Note that the default gateway for the vehicle has not yet been changed. All data packets from/to the vehicle are still processed by the previous RSU.

4.4.3 Processing the UDP-NS for Handover at the New RSU

The RSU NDP module checks its VVL-RSU with the target IPv6 address at each NS message in the UDP-NS message. If no matching entry exists, it forwards the NS message to the LBR NDP module. Otherwise it sends an NA message with an appropriate error code to the vehicle NDP module. In both case, the NS/NA messages are combined into UDP-NS/NA messages.

4.4.4 Processing the UDP-NS for Handover at the LBR

The LBR checks its VVL-LBR with the target IPv6 address at each NS message in the UDP-NS message. If no matching entry exists, it generates an NA with an appropriate error code. Otherwise, it updates the serving RSU subnet identifier field of the matching entry with the new RSU subnet identifier presented in the source IPv6 address of the UDP-NS message. It also generates an NA message with the success code. Subsequently, it combines all NA messages into one UDP-NA message and returns it to the RSU NDP module of the new RSU.

4.4.5 Processing the UDP-NA at New RSU

If there is no error code at each NA message in the UDP-NA message, the RSU NDP module creates a VVL-RSU entry for the vehicle using the target IPv6 address and link-layer address in the NA message. It then forwards the received UDP-NA to the vehicle NDP module.

4.4.6 Processing the UDP-NA at the Vehicle

When the vehicle NDP module receives the UDP-NA message, it reports the result of each NS message to the IPv6 configuration module. If the result presents handover success, it changes its default gateway to the new RSU. After this point, all packets are forwarded via the new RSU.

4.5 IPv6 Communication between Vehicles

Note that vehicles in our scheme determine a virtual link as off-link and send all IPv6 packets to their RSU although they can communicate directly.

4.5.1 Vehicles on the Same Virtual Link

A source vehicle acts as a CN. It follows step 1, 2 and 3 in Communication Phase. Then, its RSU receives the packet sent and forwards it to the LBR without any modification. Subsequently, step 4 and 5 in Communication Phase are followed. The packet is forwarded to the serving RSU of the destination vehicle using the subnet identifier of the serving RSU of the destination vehicle. Then, the serving RSU restores the subnet identifier of the packet to the zero and forwards the packet to the destination vehicle.

4.5.2 Vehicles on Different Virtual Links

A source vehicle also acts as a CN. It follows step 1, 2 and 3 in Communication Phase. When the LBR of the source vehicle receives the packet, it forwards the packet to the Internet. Through the Internet, the packet arrives at the LBR of the destination vehicle. After that, step 4 and 5 in Communication Phase are followed by the LBR and serving RSU of the destination vehicle.

5. The Proposed System Analysis

Table 1 lists symbols used for our analysis. According to [32], a stable radius of an RSU coverage should be 370 m if there is no obstruction [32]. We assume that D_{RSU} is 740 m and NL_{RSU} is 8 in the following analysis.

The average passenger vehicle length is about 4.48 m [33]. Each vehicle should maintain a safety distance with a front vehicle. According to [34], the distance should be (v − 15) meters if v is under 80 km/h. Otherwise, the distance is the same as v. We assume that the minimum safety distance is 1m. If vehicles move at under 16 km/h, they maintain the minimum safety distance.

vpm can be defined by Eq. (1) where the vehicle length is 4.48 m (the average length). In Eq. (1), vpm depends only on the safety distance since the vehicle length is a constant. As a result, vpm has the maximum value with the minimum safety distance; the value is about 0.18 if v is under 16 km/h.

Table 1 Symbols for the system analysis.

D_{RSU}	Diameter of an RSU coverage
NL_{RSU}	The number of lanes within an RSU
vpm	The number of vehicles per meter
NVL_{RSU}	The number of vehicles per lane within an RSU
NV_{RSU}	The number of vehicles within an RSU
NR_{LBR}	The number of RSUs within an LBR
NV_{LBR}	The number of vehicles within an LBR
D_{LBR}	The distance covered by an LBR
VE	The size of a VVL-LBR entry (18 octets)
MV_{LBR}	The memory used for a VVL-LBR
i_{udpns}	The UDP-NS generation interval of a vehicle
v	The speed of vehicles (km/h)
cv	The value of v without units
N_{udpns}	The number of UDP-NS messages per second at an LBR

For example, if vehicles are moving at 16 km/h, 40 km/h and 100 km/h, safety distances are 1 m, 25 m and 100 m. vpms become approximately 0.18, 0.034 and 0.01, respectively.

$$vpm = \frac{1}{(vehicle\ length + safety\ distance)} \tag{1}$$

NV_{RSU} can be calculated by Eq. (2) where the lane length is same as D_{RSU}. For example, if the vpm is 0.18 (the highest vpm), NVL_{RSU} and NV_{RSU} become up to approximately 133.2 and 1065.6, respectively.

$$\begin{aligned} NVL_{RSU} &= D_{RSU} \cdot vpm \\ NV_{RSU} &= NVL_{RSU} \cdot NL_{RSU} \end{aligned} \tag{2}$$

NV_{LBR} and D_{LBR} are calculated by Eq. (3). In our scheme, since each RSU must have its own non-zero 16-bit subnet identifier, NR_{LBR} can be up to $2^{16} - 1$. Hence, NV_{LBR} may become up to 69,834,096, and D_{LBR} can be up to 48,495.9 km.

$$\begin{aligned} NV_{LBR} &= NV_{RSU} \cdot NR_{LBR} \\ D_{LBR} &= D_{RSU} \cdot NR_{LBR} \end{aligned} \tag{3}$$

Since each vehicle creates a VVL-LBR entry and its size (VE) is 18 bytes (16-byte IPv6 address and 2-byte subnet identifier), MV_{LBR} can be calculated by Eq. (4). MV_{LBR} can be up to approximately 1.26 GB if NV_{LBR} has the maximum value (69,834,096). In addition, each vehicle generates UDP-NS messages whenever it passes through an RSU. i_{udpns} is calculated by dividing D_{RSU} with v. Using i_{udpns}, N_{udpns} can be calculated by Eq. (5). If v is 16 km/h, vpm is 0.18 and NR_{RSU} is $2^{16} - 1$, NV_{LBR} is the maximum value (69,834,096) and i_{udpns} is 166.5 seconds. Therefore, N_{udpns} becomes 419,424 messages per second.

$$\begin{aligned} MV_{LBR} &= NV_{LBR} \cdot VE \\ &= (NV_{RSU} \cdot NR_{LBR}) \cdot VE \\ &= (D_{RSU} \cdot vpm \cdot NL_{RSU} \cdot NR_{LBR}) \cdot VE \end{aligned} \tag{4}$$

Equation (4) implies that MV_{LBR} is increased by NV_{LBR} and vpm since D_{RSU}, NL_{RSU}, and VE are constants.

$$\begin{aligned} N_{udpns} &= \frac{NV_{LBR}}{i_{udpns}} = \frac{NV_{RSU} \cdot NR_{LBR}}{\frac{D_{RSU}}{v}} \\ &= \frac{D_{RSU} \cdot vpm \cdot NL_{RSU} \cdot NR_{LBR} \cdot v}{D_{RSU}} \\ &= vpm \cdot NL_{RSU} \cdot NR_{LBR} \cdot v \end{aligned} \tag{5}$$

Equation (5) implies that N_{udpns} is also increased by NR_{LBR}, vpm and v. However, as described above, vpm is related to v due to the safety distance. Therefore, $(v \cdot vpm)$ must be considered for N_{udpns} and it can be calculated by Eq. (6) and is figured in Fig. 7. If cv is greater than 16 and less than 80, the derivative of Eq. (6) with respect to cv is always negative. If cv is greater than 80, the derivative is always positive, but $(v \cdot vpm)$ converges to $\frac{5}{18}$. Therefore, $(v \cdot vpm)$ has the maximum value when v is 16 km/h and it is approximately 0.811.

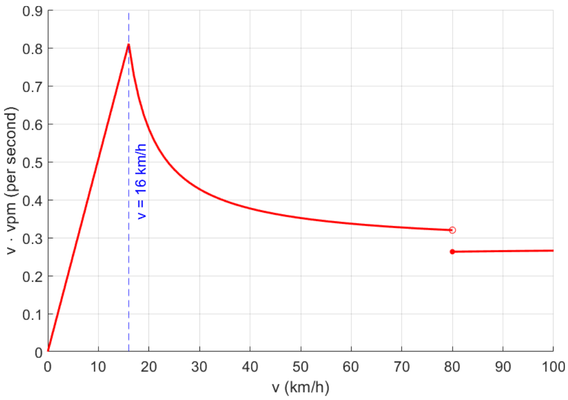
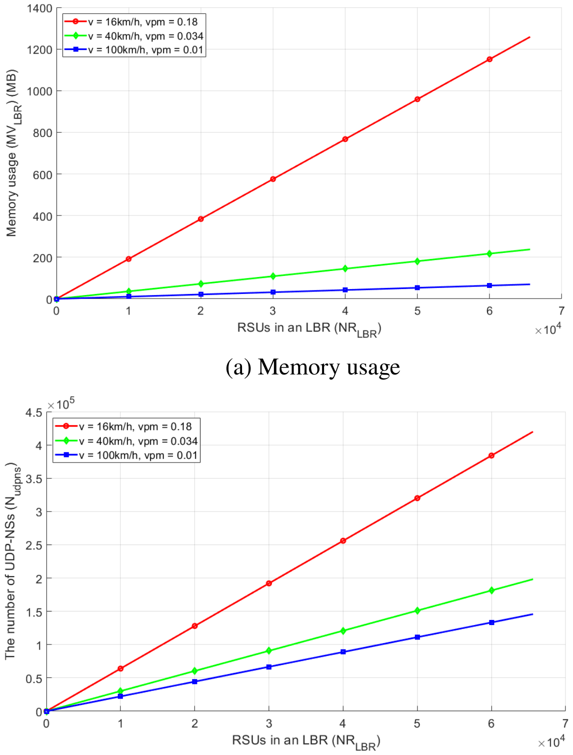


Fig. 7 $v \cdot vpm$ graph.



(a) Memory usage

(b) The number of UDP-NS processing

Fig. 8 Memory usage and the number of UDP-NS processing.

$$v \cdot vpm = \begin{cases} \frac{1}{4.48+1} \cdot \frac{1000 \cdot cv}{3600}, & cv \leq 16 \\ \frac{1}{4.48+(cv-15)} \cdot \frac{1000 \cdot cv}{3600}, & 16 < cv < 80 \\ \frac{1}{4.48+cv} \cdot \frac{1000 \cdot cv}{3600}, & cv \geq 80 \end{cases} \tag{6}$$

Figure 8 show MV_{LBR} and N_{udpns} for three cases of (v, vpm), which are (16 km/h, 0.18), (40 km/h, 0.034) and (100 km/h, 0.01). Regardless of cases, MV_{LBR} and N_{udpns} are increased by NR_{LBR}. In addition, MV_{LBR} is more rapidly increased if vpm is higher. N_{udpns} is more rapidly increased if $(v \cdot vpm)$ is higher.

As shown in Eq. (4) and Eq. (5), both MV_{LBR} and

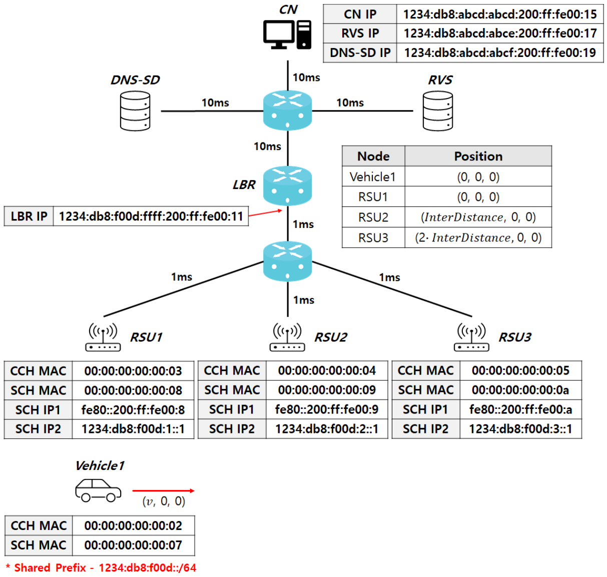
N_{udpns} are increased if either NR_{LBR} or vpm is increased. Usually NR_{LBR} is related to the coverage distance of an LBR. Mostly an LBR needs not to cover the maximum distance, 48,495.9 km. For example, only 1352 RSUs are required to cover a 1000 km highway. If (v, vpm) is (16 km/h, 0.18), an LBR requires approximately 26 MB memory and processes 8,653 UDP-NS messages per second.

Equation (2) describes that NV_{RSU} is determined by only vpm if D_{RSU} and NL_{RSU} are constants. Hence, by using methods to reduce NV_{RSU}, vpm can be reduced. One of the methods is to introduce multiple LBRs within the same coverage distance. In this method, each LBR manages a shared prefix for the coverage distance and RSUs advertise all shared prefixes assigned to the coverage distance. Each shared prefix may be advertised in round-robin fashion or included in an additional WAVE IE of the service information of an WSA. Each vehicle uses the shared prefix in an WSA it firstly receives to configure its IPv6 module. As a result, vehicles in an RSU (NV_{RSU}) are randomly distributed into multiple LBRs. It causes the expected value of vpm to be reduced without any effect to v, so both expected values of MV_{LBR} and N_{udpns} are also reduced. For example, if there are 10 LBRs at the above example and NV_{RSU} is equally distributed, each LBR requires approximately 2.6 MB memory and processes 865.3 UDP-NS messages per second.

6. Experiments

We simulated the proposed scheme with NS-3 version 3.32 to show the IPv6 service continuity between a vehicle and CN. The network topology used in the simulation is shown in Fig. 9. Each wired link had a different delay. The MAC and IPv6 addresses of each node and the delay of each link are presented in Fig. 9. The simulation parameters are listed in Table 2. The communication range of the vehicle and RSUs was set to approximately 370 m, which is a reliable communication range of VANETs where there is no obstructions [32]. The vehicle and RSUs had two WAVE interfaces. One of the interfaces was continuously tuned to the CCH. Through this interface, they exchanged WSAs which included the IPv6 routing PSID. The IPv6 routing service used the SCH1. The speed, v, of the vehicle was the 16.67 m/s (60 km/h), and the RSU inter-distance was 1000 m.

Figure 10 shows WSAs received at the vehicle via the CCH and Fig. 11 shows messages exchanged between the vehicle and RSUs via the SCH1. In Fig. 11, we omitted the IEEE 802.11 ACK frames for simplicity. The vehicle received the first WSA sent from the RSU1 at 0.000434 seconds (s). The vehicle then generated a link-local IPv6 address (fe80::2bfa:9152:9be4:930b) and a global IPv6 address (1234:db8:f00d:0:e470:8687:d7a3:8f9b) based on the advertised IPv6 prefix (1234:db8:f00d::/64). Then it registered the newly generated IPv6 addresses via the vehicle NDP module. The contents of UDP-NS and UDP-NA messages (Packet #2 and #4 in Fig. 11) are shown in Fig. 12(a) and Fig. 12(b), respectively. When the UDP-NS message was to be sent, there was no valid address at the vehicle; the source IPv6 address of the message used the unspecified address (::). Even when the destination IPv6 address of the UDP-NA message was the all-node multicast IPv6 address (FF02::1), the link-layer address in the SLLAO option of any NA message in the UDP-NA message was used instead of the link-layer multicast address. Because the registration was successful, these tentative addresses became preferred addresses. The vehicle sent a DNS update message with the registered global IPv6 address to its RVS (Packet #6 in Fig. 11).



CN IP	1234:db8:abcd:abcd:200:ff:fe00:15
RVS IP	1234:db8:abcd:abce:200:ff:fe00:17
DNS-SD IP	1234:db8:abcd:abcf:200:ff:fe00:19

Node	Position
Vehicle1	(0, 0, 0)
RSU1	(0, 0, 0)
RSU2	($InterDistance$, 0, 0)
RSU3	(2· $InterDistance$, 0, 0)

LBR IP	1234:db8:f00d:ffff:200:ff:fe00:11

CCH MAC	00:00:00:00:00:03
SCH MAC	00:00:00:00:00:08
SCH IP1	fe80::200:ff:fe00:8
SCH IP2	1234:db8:f00d:1::1

CCH MAC	00:00:00:00:00:04
SCH MAC	00:00:00:00:00:09
SCH IP1	fe80::200:ff:fe00:9
SCH IP2	1234:db8:f00d:2::1

CCH MAC	00:00:00:00:00:05
SCH MAC	00:00:00:00:00:0a
SCH IP1	fe80::200:ff:fe00:a
SCH IP2	1234:db8:f00d:3::1

CCH MAC	00:00:00:00:00:02
SCH MAC	00:00:00:00:00:07

* Shared Prefix - 1234:db8:f00d::/64

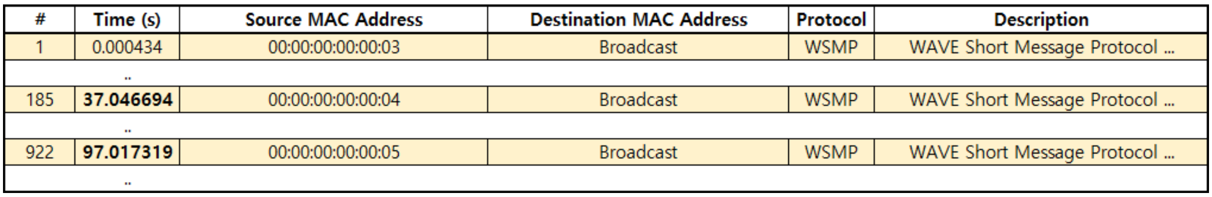
Fig. 9 Simulation environment.

Table 2 Simulation parameters.

Wireless Tx Power	16.02 dBm
Wireless Tx data rate	6 Mbps
Propagation loss model	Log-distance
Reference loss	47.86 dB (5.9 Ghz)
Path loss exponent	2 (Free-space)
Preamble detection model	Threshold (−83.5 dBm)
Wired link speed	100 Mbps
The NAI of the vehicle	nam-2@korea.ac.kr

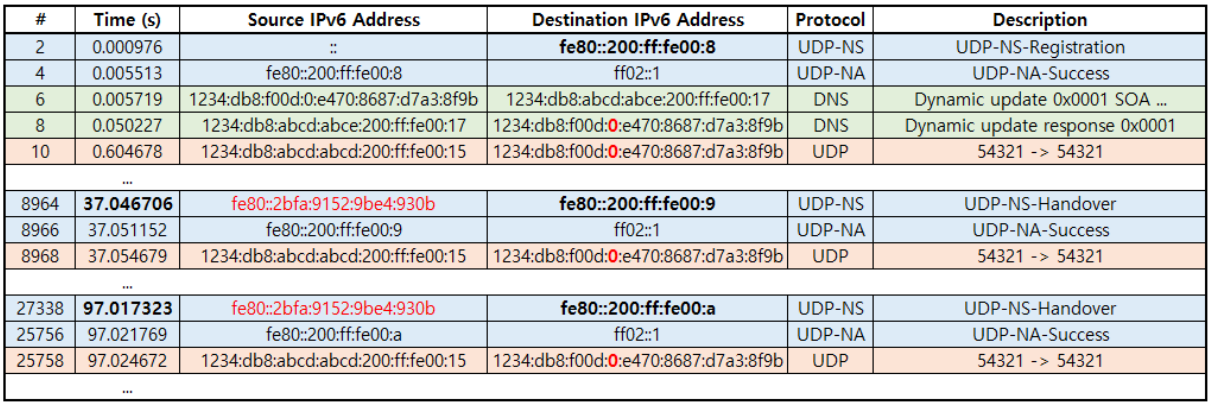
The CN started at 0.5s after the simulation started. Fig. 13 shows the messages sent or received by the CN. The CN constructed the RVS service name _rvs._udp.korea.ac.kr using the NAI of the vehicle nam-2@korea.ac.kr. Then, it exchanged DNS PTR query/response for the RVS service name with the DNS-SD server (Packet #2 and #3 in Fig. 13). The contents of the DNS PTR query and response are shown in Fig. 14(a) and Fig. 14(b), respectively. Subsequently, the CN exchanged DNS AAAA query/response with the RVS (Packet #4 and #5 in Fig. 13). Using the vehicle IPv6 address acquired from the RVS, the CN started to send UDP packets to the vehicle (Packet #6 in Fig. 13) at 0.580093 s. At 0.6004678 s, the vehicle received the first UDP message (Packet #10 of Fig. 11) sent by the CN.

At 37.046694 s, the vehicle received a new WSA sent by the RSU2 via its CCH interface. As the prefix advertised



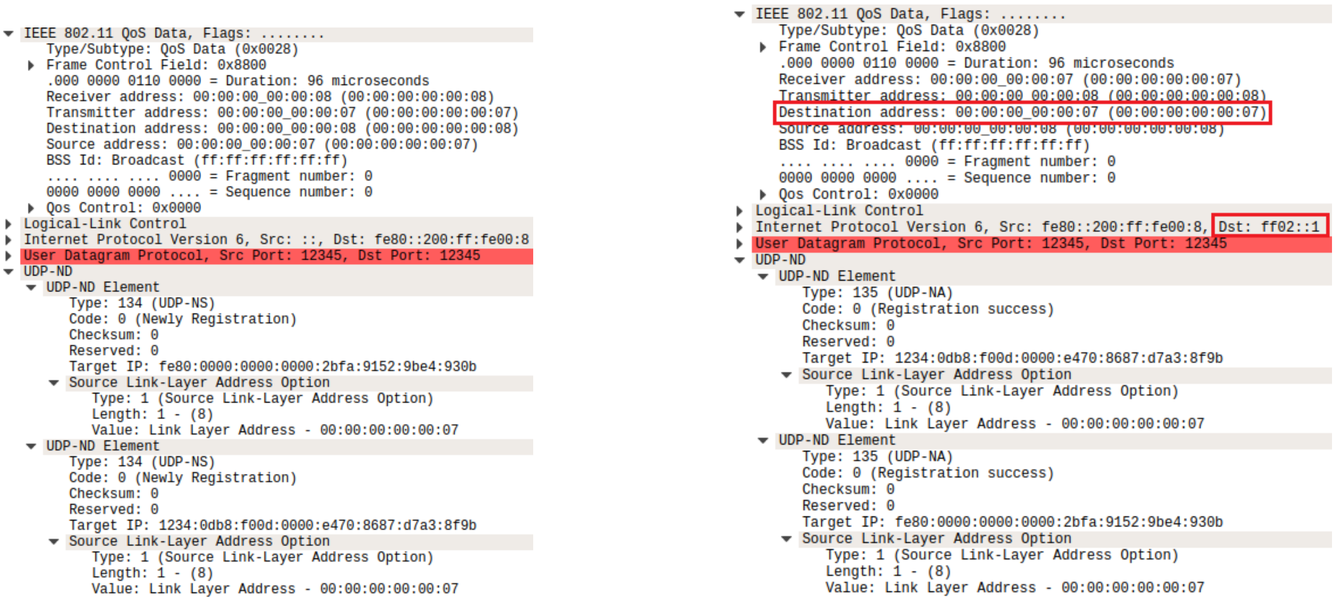
#	Time (s)	Source MAC Address	Destination MAC Address	Protocol	Description
1	0.000434	00:00:00:00:00:03	Broadcast	WSMP	WAVE Short Message Protocol ...
	..				
185	**37.046694**	00:00:00:00:00:04	Broadcast	WSMP	WAVE Short Message Protocol ...
	..				
922	**97.017319**	00:00:00:00:00:05	Broadcast	WSMP	WAVE Short Message Protocol ...
	..				

Fig. 10 Captured packets on the CCH interface of the vehicle.



#	Time (s)	Source IPv6 Address	Destination IPv6 Address	Protocol	Description
2	0.000976	::	**fe80::200:ff:fe00:8**	UDP-NS	UDP-NS-Registration
4	0.005513	fe80::200:ff:fe00:8	ff02::1	UDP-NA	UDP-NA-Success
6	0.005719	1234:db8:f00d:0:e470:8687:d7a3:8f9b	1234:db8:abcd:abce:200:ff:fe00:17	DNS	Dynamic update 0x0001 SOA ...
8	0.050227	1234:db8:abcd:abce:200:ff:fe00:17	1234:db8:f00d:0:e470:8687:d7a3:8f9b	DNS	Dynamic update response 0x0001
10	0.604678	1234:db8:abcd:abcd:200:ff:fe00:15	1234:db8:f00d:0:e470:8687:d7a3:8f9b	UDP	54321 -> 54321
	...				
8964	**37.046706**	fe80::2bfa:9152:9be4:930b	**fe80::200:ff:fe00:9**	UDP-NS	UDP-NS-Handover
8966	37.051152	fe80::200:ff:fe00:9	ff02::1	UDP-NA	UDP-NA-Success
8968	37.054679	1234:db8:abcd:abcd:200:ff:fe00:15	1234:db8:f00d:0:e470:8687:d7a3:8f9b	UDP	54321 -> 54321
	...				
27338	**97.017323**	fe80::2bfa:9152:9be4:930b	**fe80::200:ff:fe00:a**	UDP-NS	UDP-NS-Handover
25756	97.021769	fe80::200:ff:fe00:a	ff02::1	UDP-NA	UDP-NA-Success
25758	97.024672	1234:db8:abcd:abcd:200:ff:fe00:15	1234:db8:f00d:0:e470:8687:d7a3:8f9b	UDP	54321 -> 54321
	...				

Fig. 11 Captured packets on the SCH interface of the vehicle.



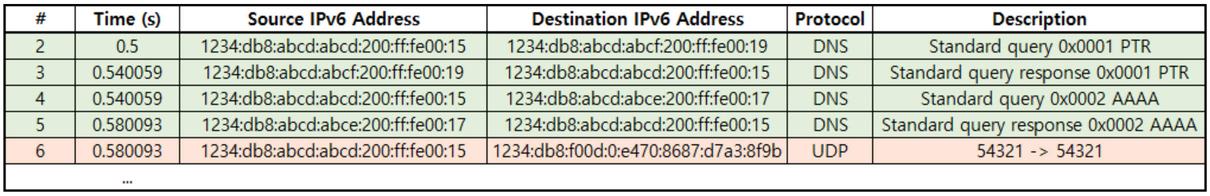
```
▼ IEEE 802.11 QoS Data, Flags: ........
    Type/Subtype: QoS Data (0x0028)
  ▶ Frame Control Field: 0x8800
    .000 0000 0110 0000 = Duration: 96 microseconds
    Receiver address: 00:00:00_00:00:08 (00:00:00:00:00:08)
    Transmitter address: 00:00:00_00:00:07 (00:00:00:00:00:07)
    Destination address: 00:00:00_00:00:08 (00:00:00:00:00:08)
    Source address: 00:00:00_00:00:07 (00:00:00:00:00:07)
    BSS Id: Broadcast (ff:ff:ff:ff:ff:ff)
    .... .... .... 0000 = Fragment number: 0
    0000 0000 0000 .... = Sequence number: 0
  ▶ Qos Control: 0x0000
▶ Logical-Link Control
▶ Internet Protocol Version 6, Src: ::, Dst: fe80::200:ff:fe00:8
▶ User Datagram Protocol, Src Port: 12345, Dst Port: 12345
▼ UDP-ND
  ▼ UDP-ND Element
      Type: 134 (UDP-NS)
      Code: 0 (Newly Registration)
      Checksum: 0
      Reserved: 0
      Target IP: fe80:0000:0000:0000:2bfa:9152:9be4:930b
    ▼ Source Link-Layer Address Option
        Type: 1 (Source Link-Layer Address Option)
        Length: 1 - (8)
        Value: Link Layer Address - 00:00:00:00:00:07
  ▼ UDP-ND Element
      Type: 134 (UDP-NS)
      Code: 0 (Newly Registration)
      Checksum: 0
      Reserved: 0
      Target IP: 1234:0db8:f00d:0000:e470:8687:d7a3:8f9b
    ▼ Source Link-Layer Address Option
        Type: 1 (Source Link-Layer Address Option)
        Length: 1 - (8)
        Value: Link Layer Address - 00:00:00:00:00:07
```

(a) UDP-NS for registration (Packet #2)

```
▼ IEEE 802.11 QoS Data, Flags: ........
    Type/Subtype: QoS Data (0x0028)
  ▶ Frame Control Field: 0x8800
    .000 0000 0110 0000 = Duration: 96 microseconds
    Receiver address: 00:00:00_00:00:07 (00:00:00:00:00:07)
    Transmitter address: 00:00:00_00:00:08 (00:00:00:00:00:08)
    Destination address: 00:00:00_00:00:07 (00:00:00:00:00:07)
    Source address: 00:00:00_00:00:08 (00:00:00:00:00:08)
    BSS Id: Broadcast (ff:ff:ff:ff:ff:ff)
    .... .... .... 0000 = Fragment number: 0
    0000 0000 0000 .... = Sequence number: 0
  ▶ Qos Control: 0x0000
▶ Logical-Link Control
▶ Internet Protocol Version 6, Src: fe80::200:ff:fe00:8, Dst: ff02::1
▶ User Datagram Protocol, Src Port: 12345, Dst Port: 12345
▼ UDP-ND
  ▼ UDP-ND Element
      Type: 135 (UDP-NA)
      Code: 0 (Registration success)
      Checksum: 0
      Reserved: 0
      Target IP: 1234:0db8:f00d:0000:e470:8687:d7a3:8f9b
    ▼ Source Link-Layer Address Option
        Type: 1 (Source Link-Layer Address Option)
        Length: 1 - (8)
        Value: Link Layer Address - 00:00:00:00:00:07
  ▼ UDP-ND Element
      Type: 135 (UDP-NA)
      Code: 0 (Registration success)
      Checksum: 0
      Reserved: 0
      Target IP: fe80:0000:0000:0000:2bfa:9152:9be4:930b
    ▼ Source Link-Layer Address Option
        Type: 1 (Source Link-Layer Address Option)
        Length: 1 - (8)
        Value: Link Layer Address - 00:00:00:00:00:07
```

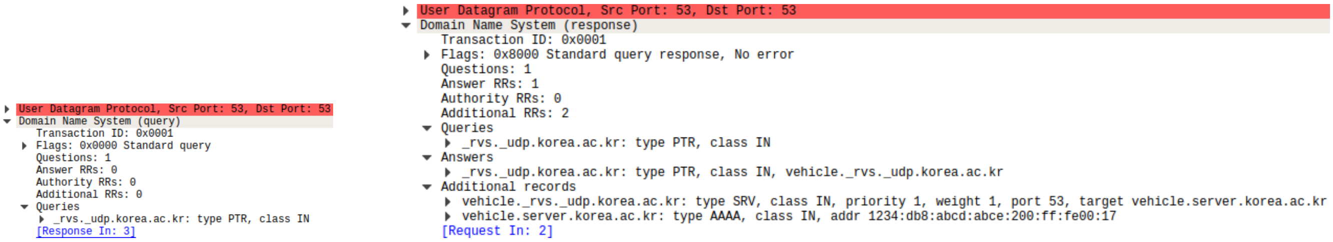
(b) UDP-NA for registration success (Packet #4)

Fig. 12 Captured UDP-NS and UDP-NA messages in Fig. 11.



#	Time (s)	Source IPv6 Address	Destination IPv6 Address	Protocol	Description
2	0.5	1234:db8:abcd:abcd:200:ff:fe00:15	1234:db8:abcd:abcf:200:ff:fe00:19	DNS	Standard query 0x0001 PTR
3	0.540059	1234:db8:abcd:abcf:200:ff:fe00:19	1234:db8:abcd:abcd:200:ff:fe00:15	DNS	Standard query response 0x0001 PTR
4	0.540059	1234:db8:abcd:abcd:200:ff:fe00:15	1234:db8:abcd:abce:200:ff:fe00:17	DNS	Standard query 0x0002 AAAA
5	0.580093	1234:db8:abcd:abce:200:ff:fe00:17	1234:db8:abcd:abcd:200:ff:fe00:15	DNS	Standard query response 0x0002 AAAA
6	0.580093	1234:db8:abcd:abcd:200:ff:fe00:15	1234:db8:f00d:0:e470:8687:d7a3:8f9b	UDP	54321 -> 54321
	...				

Fig. 13 Captured packets on the interface of the CN.



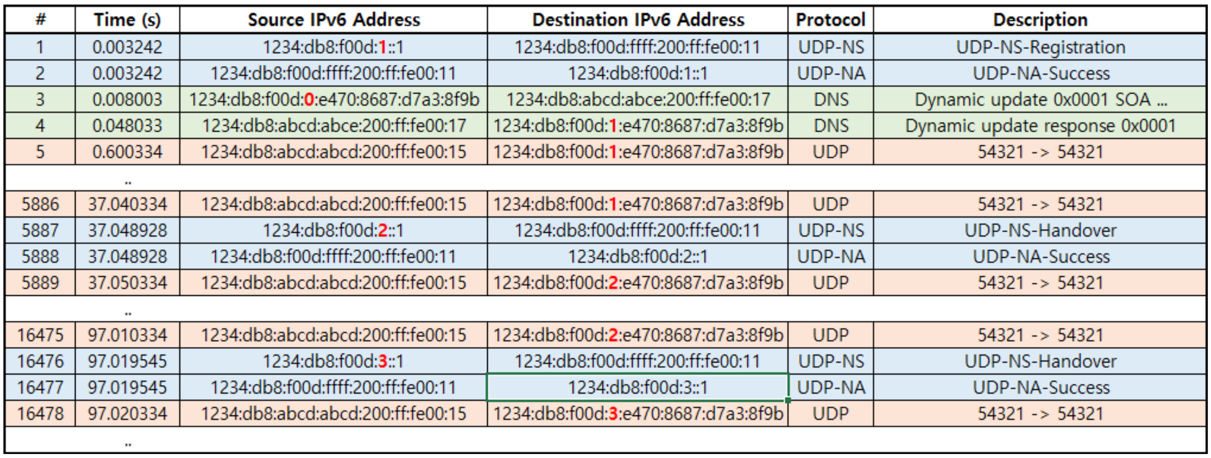
```
▶ User Datagram Protocol, Src Port: 53, Dst Port: 53
▼ Domain Name System (query)
    Transaction ID: 0x0001
  ▶ Flags: 0x0000 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  ▼ Queries
    ▶ _rvs._udp.korea.ac.kr: type PTR, class IN
    [Response In: 3]
```

(a) DNS Query (Packet #2)

```
▶ User Datagram Protocol, Src Port: 53, Dst Port: 53
▼ Domain Name System (response)
    Transaction ID: 0x0001
  ▶ Flags: 0x8000 Standard query response, No error
    Questions: 1
    Answer RRs: 1
    Authority RRs: 0
    Additional RRs: 2
  ▼ Queries
    ▶ _rvs._udp.korea.ac.kr: type PTR, class IN
  ▼ Answers
    ▶ _rvs._udp.korea.ac.kr: type PTR, class IN, vehicle._rvs._udp.korea.ac.kr
  ▼ Additional records
    ▶ vehicle._rvs._udp.korea.ac.kr: type SRV, class IN, priority 1, weight 1, port 53, target vehicle.server.korea.ac.kr
    ▶ vehicle.server.korea.ac.kr: type AAAA, class IN, addr 1234:db8:abcd:abce:200:ff:fe00:17
    [Request In: 2]
```

(b) DNS Response (Packet #3)

Fig. 14 Captured DNS messages for DNS-SD in Fig. 13.



#	Time (s)	Source IPv6 Address	Destination IPv6 Address	Protocol	Description
1	0.003242	1234:db8:f00d:1::1	1234:db8:f00d:ffff:200:ff:fe00:11	UDP-NS	UDP-NS-Registration
2	0.003242	1234:db8:f00d:ffff:200:ff:fe00:11	1234:db8:f00d:1::1	UDP-NA	UDP-NA-Success
3	0.008003	1234:db8:f00d:0:e470:8687:d7a3:8f9b	1234:db8:abcd:abce:200:ff:fe00:17	DNS	Dynamic update 0x0001 SOA ...
4	0.048033	1234:db8:abcd:abce:200:ff:fe00:17	1234:db8:f00d:1:e470:8687:d7a3:8f9b	DNS	Dynamic update response 0x0001
5	0.600334	1234:db8:abcd:abcd:200:ff:fe00:15	1234:db8:f00d:1:e470:8687:d7a3:8f9b	UDP	54321 -> 54321
	..				
5886	37.040334	1234:db8:abcd:abcd:200:ff:fe00:15	1234:db8:f00d:1:e470:8687:d7a3:8f9b	UDP	54321 -> 54321
5887	37.048928	1234:db8:f00d:2::1	1234:db8:f00d:ffff:200:ff:fe00:11	UDP-NS	UDP-NS-Handover
5888	37.048928	1234:db8:f00d:ffff:200:ff:fe00:11	1234:db8:f00d:2::1	UDP-NA	UDP-NA-Success
5889	37.050334	1234:db8:abcd:abcd:200:ff:fe00:15	1234:db8:f00d:2:e470:8687:d7a3:8f9b	UDP	54321 -> 54321
	..				
16475	97.010334	1234:db8:abcd:abcd:200:ff:fe00:15	1234:db8:f00d:2:e470:8687:d7a3:8f9b	UDP	54321 -> 54321
16476	97.019545	1234:db8:f00d:3::1	1234:db8:f00d:ffff:200:ff:fe00:11	UDP-NS	UDP-NS-Handover
16477	97.019545	1234:db8:f00d:ffff:200:ff:fe00:11	1234:db8:f00d:3::1	UDP-NA	UDP-NA-Success
16478	97.020334	1234:db8:abcd:abcd:200:ff:fe00:15	1234:db8:f00d:3:e470:8687:d7a3:8f9b	UDP	54321 -> 54321
	..				

Fig. 15 Captured packets on the interface of the LBR towards the WAVE network.

by the WRA in the WSA did not change, the vehicle notified its movement to the RSU2 (Packet #8964 and #8966 in Fig. 11). In this case, the vehicle had its link-local address (fe80::2bfa:9152:9be4:930b); therefore, the vehicle used it as the source IPv6 address of an UDP-NS message for the handover. The global IPv6 address was not changed at this time and no DNS update to the RVS was required. After the handover success at 37.051152 s, the vehicle started to successfully receive UDP packets sent by the CN (from Packet #8968 in Fig. 11) via the RSU2.

Figure 15 shows the messages exchanged between RSUs and the LBR in the WAVE access network. The LBR received the UDP-NS message for the address registration (Packet #1 in Fig. 15) from the RSU1 at 0.003242s. The RSU1 used the source IPv6 address of the message which contains its subnet identifier (1234:db8:f00d:1::1). When the LBR processed each NS message in the UDP-NS message, no address duplication is detected; therefore, it generates NA messages with the code success. All NAs are combined into one UDP-NA message (Packet #2 in Fig. 15) and the UDP-NA message was sent to the RSU1. After the RSU1 received the UDP-NA message, it created new VVL-RSU entries using the UDP-NA message and forwarded the message using the link-layer address included in the SLLAO option of any NA message in the UDP-NA message. Then, data packets destined for the vehicle were correctly delivered to the RSU1 by inserting the RSU1 subnet identifier (1) into the destination IPv6 address of the packets (from Packet #4 to #5886 in Fig. 15). Whenever the RSU1 received a packet destined for the vehicle, it replaced its subnet identifier of the destination IPv6 address of the packet with zero and forwarded it to the vehicle (from Packet #8 in Fig. 11).

At 37.048928, the LBR received an UDP-NS for the handover notification from the RSU2 (Packet #5887 in Fig. 15). After the LBR processed the UDP-NS message, it changed the serving RSU of the vehicle from the RSU1 to the RSU2. Then, the LBR sent the UDP-NA message (Packet #5888 in Fig. 15) to the RSU2. The RSU2 created new VVL-RSU entries using the UDP-NA message and relayed the message to the vehicle (Packet #8966 in Fig. 11). Then, data packets destined for the vehicle were correctly delivered to the RSU2 by inserting the RSU2 subnet identifier (2) into the destination IPv6 address of the packets (from Packet #5889 to #16475 in Fig. 15). Whenever the RSU2 received a packet destined for the vehicle, it replaced its subnet identifier (2) of the destination IPv6 address of the packet with zero and forwarded it to the vehicle (from Packet #8968 in Fig. 11).

We also simulated overlapped/non-overlapped RSU coverage cases with 500 m and 1000 m inter-RSU distances because the radius of the RSU coverage was approximately 370 m in our simulation. The results are illustrated in Fig. 16. In the overlapped case, no packets were lost during the handover period. However, in the non-overlapped case, some packet were lost during the handover period because the vehicle could not receive packets until it moved into the coverage of a new RSU which belongs to the same virtual link.

7. Conclusion and Future Work

We proposed an identifier locator separation protocol for the shared prefix model over the IEEE WAVE IPv6 network and demonstrated its applicability to the WAVE network by running various simulations. In the shared prefix model, vehicles do not change their IPv6 addresses within the shared prefix domain. In addition, the IPv6 addresses are based on the stable IID. The IPv6 addresses are decoupled from the link-layer address of an interface. The protocol can coexist with the standard NDP since it uses UDP encapsulation instead of ICMPv6. This scheme does not use any link-level multicast to maximize wireless channel efficiency.

In our scheme, some packets are lost during handover period if the RSU coverages do not overlap, as shown in Fig. 16. This problem can be easily remedied by sending one extra NS packet by the vehicle to the LBR, if it does not detect new RSU before it leaves the current RSU coverage. The LBR holds its packets until it receives another NS message from the vehicle. Another problem is that all IPv6 packets pass their RSUs even though vehicles are reachable from each other. It occurs route inefficiency. To handle this problem, route optimization algorithms may be introduced.

There are several security considerations for our scheme. First of all, if a sender of UDP-ND messages is not authenticated, an attacker can exhaust the 64-bit IID space over a virtual link by sending false UDP-NS registration messages. In addition, it can also change the current RSU for a victim vehicle by sending false UDP-NS handover messages. Secondly, if integrity of UDP-ND messages is not guaranteed, a man-in-the-middle (MitM) attacker can disturb a vehicle to register its configured IPv6 address into the LBR by modifying the target address field in UDP-NS messages. In addition, the attacker can maliciously change the current RSU for a victim vehicle by modifying the source IPv6 address of UDP-NS messages forwarded by RSUs. Finally, if the attacker collects UDP-NS handover messages and replays them, the current RSU of a victim vehicle is changed to the previous RSU. To handle these considerations, a Secure Neighbor Discovery (SEND) [35]-based NDP options which exploits 1609.2 certificates may be introduced. They are used to authenticate a vehicle and an RSU, guarantee integrity of UDP-ND message contents and prevent replaying messages. In addition, IPsec with pre-configured security associations may be used between an RSU and the LBR, in order to authenticate each other and protect the source IPv6 address of UDP-NS messages against an MitM attacker.

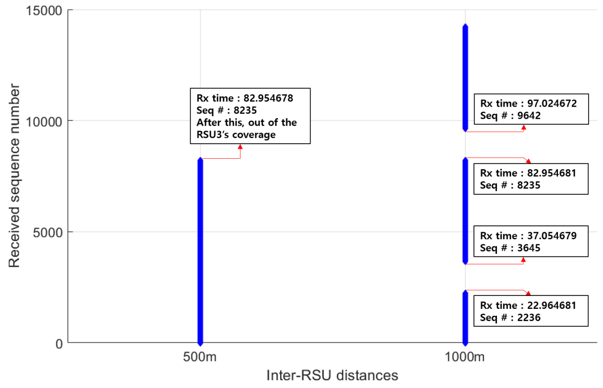


Fig. 16 Received sequence numbers at the vehicle for 500 m and 1000 m inter-RSU distances.

Acknowledgments

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Ministry of Science and ICT (MSIT) (No. 2021-0-01774, Development of IEC 62443 based Smart Factory Security Internalization and Embedded Device Security Technology) and partially supported by College of Informatics, Korea University.

References

[1] J. Jeong, B. Mugabarigira, Y. Shen, and Z. Xiang, "Vehicular mobility management for IP-based vehicular networks," IETF draft, draft-jeong-ipwave-vehicular-mobility-management-05, Feb. 2021.

[2] J. Jeong, Y. Shen, and Z. Xiang, "Vehicular neighbor discovery for IP-based vehicular networks," IETF draft, draft-jeong-ipwave-vehicular-neighbor-discovery-11, Feb. 2021.

[3] R. Moskowitz, T. Heer, P. Jokela, and T. Henderson, "Host identity protocol version 2 (HIPv2)," IETF, RFC 7401, April 2015.

[4] D. Farinacci, V. Fuller, D. Meyer, and D. Lewis, "The locator/ID separation protocol (LISP)," IETF, RFC 6830, Jan. 2013.

[5] R.J. Atkinson and S.N. Bhatti, "Identifier-locator network protocol (ILNP) architectural description," IETF, RFC 6740, Nov. 2012.

[6] P. Mockapetris, "Domain names — Concepts and facilities," IETF, RFC 1034, Nov. 1987.

[7] R.J. Atkinson, S.N. Bhatti, and S. Rose, "DNS resource records for the identifier-locator network protocol (ILNP)," IETF, RFC 6742, Nov. 2012.

[8] T. Henderson, C. Vogt, and J. Arkko, "Host mobility with the host identity protocol," IETF, RFC 8046, Feb. 2017.

[9] D. Farinacci, D. Lewis, D. Meyer, and C. White, "LISP mobile node," IETF draft, draft-ietf-lisp-mn-10, Aug. 2021.

[10] R.J. Atkinson and S.N. Bhatti, "ICMP locator update message for the identifier-locator network protocol for IPv6 (ILNPv6)," IETF, RFC 6743, Nov. 2012.

[11] K. Sun and Y. Kim, "Considerations for ID/location separation protocols in IPv6-based vehicular networks," IETF draft, draft-kjsun-ipwave-id-loc-separation-03, Oct. 2020.

[12] T. Narten, E. Nordmark, W. Simpson, and H. Soliman, "Neighbor discovery for IP version 6 (IPv6)," IETF, RFC 4861, Sept. 2007.

[13] A. Dekok, "The network access identifier," IETF, RFC 7542, May 2015.

[14] S. Cheshire and M. Krochmal, "DNS-based service discovery," IETF, RFC 6763, Feb. 2013.

[15] F. Gont, "A method for generating semantically opaque interface identifiers with IPv6 stateless address autoconfiguration (SLAAC)," IETF, RFC 7217, April 2014.

[16] F. Gont, A. Cooper, D. Thaler, and W. Liu, "Recommendation on stable IPv6 interface identifiers," IETF, RFC 8064, Feb. 2017.

[17] The Network Simulator 3 website (2022, Aug. 24), [Online], Available: https://www.nsnam.org/

[18] IEEE Standard for Wireless Access in Vehicular Environments (WAVE) — Networking, IEEE, New York, NY, USA, 2020.
[19] IEEE Guide for Wireless Access in Vehicular Environments (WAVE) Architecture, IEEE, New York, NY, USA, 2019.
[20] SAE International, "V2X communications message set dictionary," SAE International, Warrendale, PA, USA, July 2020.
[21] S. Cespedes, N. Lu, and X. Shen, "VIP-WAVE: On the feasibility of IP communications in 802.11p vehicular networks," IEEE Trans. Intell. Transp. Syst., vol.14, no.1, pp.82–97, July 2012.
[22] J. Laganier and L. Eggert, "Host identity protocol (HIP) rendezvous extension," IETF, RFC 8004, Oct. 2016.
[23] D. Lewis, D. Meyer, D. Farinacci, and V. Fuller, "Interworking between Locator/ID separation protocol (LISP) and non-LISP sites," IETF, RFC 6832, Jan. 2013.
[24] P. Vixie, S. Thomson, Y. Rekhter, and J. Bound, "Dynamic updates in the domain name system (DNS UPDATE)," IETF, RFC 2136, April 1997.
[25] E. Baccelli, T.H. Clausen, and R. Wakikawa, "IPv6 operation for WAVE — Wireless access in vehicular environments," 2010 IEEE Vehicular Networking Conference (VNC), Dec. 2010.
[26] E. Baccelli and M. Townsley, "IP addressing model in ad hoc networks," IETF, RFC 5889, Sept. 2010.
[27] P. Srisuresh and K. Egevang, "Traditional IP network address translator (traditional NAT)," IETF, RFC 3022, Jan. 2001.
[28] M. Petit-Huguenin, G. Salgueiro, J. Rosenberg, D. Wing, R. Mahy, and P. Matthews, "Session traversal utilities for NAT (STUN)," IETF, RFC 8489, Feb. 2020.
[29] R. Kawashima and H. Matsuo, "Non-tunneling edge-overlay model using openflow for cloud datacenter networks," 2013 IEEE 5th International Conference on Cloud Computing Technology and Science, Dec. 2013.
[30] S. Gundavelli, M. Townsley, O. Troan, and W. Dec, "Address mapping of IPv6 multicast packets on ethernet," IETF, RFC 6085, Jan. 2011.
[31] S. Thomson, T. Narten, and T. Jinmei, "IPv6 stateless address autoconfiguration," IETF, RFC 4862, Sept. 2007.
[32] R. Meireles, M. Boban, P. Steenkiste, O. Tonguz, and J. Barros, "Experimental study on the impact of vehicular obstructions in VANETs," 2010 IEEE Vehicular Networking Conference. (VNC), Dec. 2010.
[33] The Way company website (2022, Aug. 24), [Online], Available: https://www.way.com/blog/average-car-length/
[34] The KoROAD website (2022, Aug. 24), [Online], Available: https://www.koroad.or.kr/kp_web/knCarSafe1-03.do
[35] J. Arkko, J. Kempf, B. Zill, and P. Nikander, "SEcure neighbor discovery (SEND)," IETF, RFC 3971, March 2005.

Sangjin Nam received the B.S. degree in computer science from Korea University, Seoul, South Korea, in 2020, where he is currently pursuing the Ph.D. degree in computer science and engineering. His research interests include future Internet, vehicular network, QoS, mobility protocol, and software-defined networking.

Sung-Gi Min received the B.S. degree in computer science from Korea University, Seoul, South Korea, in 1988, and the M.S. and Ph.D. degrees in computer science from the University of London in 1989 and 1993, respectively. From 1994 to 2000, he was with the LG Information and Communication Research Center, and from 2000 to 2001, he was a Professor with the Department of Computer Engineering, Dongeui University, Busan, South Korea. Since 2001, he has been a Professor with the Department of Computer Science and Engineering, Korea University. His research interests include wired/wireless communication networks, and he is interested in mobility protocols, network architectures, QoS, and mobility management in future networks.