Qualcomm
TECHNOLOGIES, INC

IPv6 Implementation and Configuration

80-VD229-1 E

December 23, 2013

Submit technical questions at:
https://support.cdmatech.com/

Confidential and Proprietary – Qualcomm Technologies, Inc.

Qualcomm Technologies, Inc.
5775 Morehouse Drive
San Diego, CA 92121
U.S.A.

Contents

Figures

Tables

Revision history

Revision	Date	Description
A	Oct 2006	Initial release
B	May 2007	Revised Sections 3.5.1, 3.5.2, and 3.6.1
C	Jan 2010	Added Chapters 7, 8, and 9
D	Jul 2010	Updated Tables 1-2, 3-1, and 3-3
E	Dec 2013	Changed document title and type; updated Table 1-1 and Section 3.7.3

1 Introduction

1.1 Purpose

This document provides information on the Qualcomm Technologies, Inc. (QTI) IPv6 implementation, including supported features, limitations, configuration parameters, and IPv6 session setup and negotiation.

NOTE: QTI does not currently support Mobile IPv6.

1.2 Scope

This document is intended for licensees, infrastructure vendors, and mobile operators interested in the features and behavior of the QTI IPv6 implementation.

1.3 Conventions

Function declarations, function names, type declarations, and code samples appear in a different font, e.g., `#include`.

Code variables appear in angle brackets, e.g., `<number>`.

Shading indicates content that has been added or changed in this revision of the document.

1.4 References

Reference documents are listed in Table 1-1. Reference documents that are no longer applicable are deleted from this table; therefore, reference numbers may not be sequential.

Table 1-1 Reference documents and standards

Ref.	Document	
Qualcomm Technologies		
Q1	*Application Note: Software Glossary for Customers*	CL93-V3077-1
Q2	*Data Services API Interface Specification and Operational Description*	80-V6415-1
Q3	*AMSS Support for Robust Header Compression (ROHC)*	80-VD208-1
Q4	*Qualcomm MSM™ Interface (QMI) Architecture*	80-VB816-1
Q5	*RM Network (RmNet) Interface*	80-VT270-1
Q6	*QMI Wireless Data Service Spec*	80-VB816-5
Standards		
S1	*IP version 6 addressing architecture*	RFC 2373

Ref.	Document	
S2	*IP Encapsulating Security Payload (ESP)*	RFC 2406
S3	*Internet Protocol, Version 6 (IPv6) Specification*	RFC 2460
S4	*Neighbor Discovery for IP Version 6 (IPv6)*	RFC 2461
S5	*IPv6 Stateless Address Autoconfiguration*	RFC 2462
S6	*Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*	RFC 2463
S7	*Basic Socket Interface Extensions for IPv6*	RFC 2553
S8	*IP Version 6 over PPP*	RFC 2472
S9	*Privacy Extensions for stateless address auto configuration for IPv6*	RFC 3041
S10	*Basic socket interface for IPv6*	RFC 3493
S11	*CDMA 2000 Wireless Network IP Standard*	TIA IS-835C
S12	*Internet Protocol Version 6 (IPv6) for Some Second and Third Generation Cellular Hosts*	RFC 3316
S13	*3rd Generation Partnership Project Technical Specification*	3GPP2 TS 23.060
S14	*E-UTRAN – eHRPD Connectivity and Interworking: Core Network Aspects*	3GPP2 X.S0057-0 v1.0
Resources		
R1	Stevens, W.R. *TCP/IP Illustrated, Volume I: The Protocols*	Addison-Wesley (1994)
R2	Stevens, W.R. *UNIX® Network Programming*	Prentice-Hall (1990)

1.5 Technical assistance

For assistance or clarification on information in this document, submit a case to QTI at https://support.cdmatech.com/.

If you do not have access to the CDMATech Support Service website, register for access or send email to support.cdmatech@qti.qualcomm.com.

1.6 Acronyms

For definitions of terms and abbreviations, see [Q1].

2 Introduction to IPv6

2.1 IPv6 addressing and architecture

2.1.1 IPv6 header format

Figure 2-1 illustrates the new IPv6 header format for the base IPv6 header. It has been improved over the IPv4 header in order to streamline routing and reduce the number of header fields.

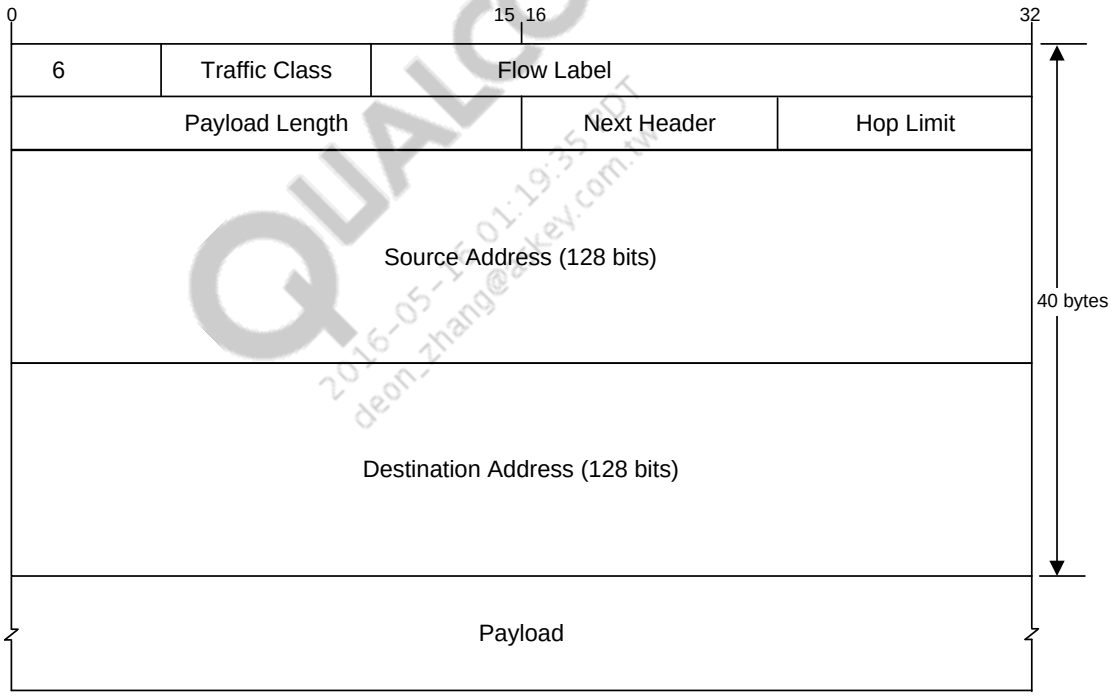

Figure 2-1 IPv6 header format

The various fields are:

- Version (4 bits) – Protocol version number = 6
- Traffic Class (8 bits) – Classes/priorities of packets; replaces the TOS field in an IPv4 header
- Flow Label (20 bits) – Per-flow classification; ultimately to be used for QoS type classifications
- Payload Length (16 bits) – Length of the payload, including any extension headers that may be present; similar to the total payload field in an IPv4 header

- Next Header (8 bits) – Identifies the header immediately following the IPv6 header (includes extension headers: hop by hop, routing, fragmentation, etc.); replaces the protocol field in an IPv4 header
- Hop limit (8 bits) – Number of routing hops before expiration; replaces TTL field in an IPv4 header
- Source Address (128 bits) – Packet originator's address
- Destination Address (128 bits) – Intended recipient's address; may not be the ultimate recipient if a routing header is present

The IPv6 addresses are formed by combination of prefix and interface ID and are represented in the following format:

2002:c023:9c17:314:bceb:acb9:5d51:4045

The IPv6 prefix is represented by the first 64 bits (2002:c023:9c17:314) and can also be represented in the format 2002:c023:9c17:314::/64.

The IID is represented by the last 64 bits (bceb:acb9:5d51:4045).

In IPv6, unlike IPv4, there is a clear separation of "who you are" (interface ID) vs. "who you are connected to" (routing prefix). The prefix is used for routing and subnet information, whereas the IID is used for a node.

Two types of prefixes are typically used by the mobile, the global and link-local prefix. The global address is represented by the first 16 bits set as 0x2001 or 0x2002, while a link local address is represented by first 16 bits set as 0xFE80. A packet with a global scoped source address can be routed globally (anywhere on the IPv6 portion of the Internet). A packet with link local source address can be routed only onto the link to which the node is connected (1 hop).

Refer to the technology-specific sections for more information on how the address is generated for different technologies.

2.1.2 IPv6 extension headers

IPv6 supports optional headers in addition to the mandatory base header. These optional headers are called extension headers. Following are the extension headers defined by IPv6 [S3].

2.1.2.1 Routing header

The routing header is used to specify a particular route that the packet should travel to its destination. All IPv6 routers are required to look at the routing header in order to send it along the correct path. When the packet is examined, the *segments left* field should be decremented by 1. When the packet reaches its destination the *segments left* should be zero. The AMSS IPv6 implementation supports this header only on the receive path and will not generate it for packets it sends. See Section 3.1.3 for information on the AMSS implementation details for IPv6 routing headers.

2.1.2.2 Hop-by-hop header

The hop-by-hop header is used to carry some optional information that must be examined by every node on the way to the packet's destination. Optional information that can be contained within the header can be padding, jumbo payload, or router alert options. Padding options are used to specify to the routers how many bytes of padding should be added to the hop-by-hop or destination options headers to ensure they fall on 8-byte boundaries. The jumbo payload option allows nodes to send packets larger than 2^{16} bytes (up to 2^{32}) called jumbograms. Because the IPv6 base header length field is only 16 bits in length, when the jumbo payload option is used, the jumbo payload length field (32 bits) in the hop-by-hop header will indicate the length of the IPv6 packet payload. The final option supported by the hop-by-hop header is the router alert option. This option indicates the router must perform additional processing. It is primarily used for Multicast Listener Discovery (MLD) and the Resource Reservation Protocol (RSVP).

2.1.2.3 Destination options header

The destination options header specifies information that must be processed at the destination nodes. These nodes can either be along the packets path to the final destination or at the final destination itself. If the destination options header is located before the routing header, every node along the path to the final destination will process the destination options header. If the header is located after the routing header, only the final destination will process the header.

Four options are provided through the destination options header: Binding Update, Binding Acknowledgement, Binding Request, and Home Address. All of these are utilized by Mobile IPv6.

2.1.2.4 Fragmentation header

The fragmentation header is for IPv6 packet fragmentation and reassembly. It is used to provide reassembly information to the final destination. Only source nodes are allowed to fragment an IPv6 packet. Routers are not allowed to fragment IPv6 packets and, therefore, do not process the fragmentation headers.

An IPv6 packet can be considered to have two parts; one is the fragmentable part and the other is the unfragmentable part.

The fragmentable part consists of IPv6 payload, authentication headers, and encapsulating security payload headers. This information is to be used only by the destination node and has no relevance to routers or intermediate nodes. Therefore, it is acceptable to fragment this information.

Unfragmentable parts consist of information that each router must look at. This includes the IPv6 header, hop by hop header, destination options header, and the routing header. This information must be present before the fragmentation header in every fragment so that routers can properly handle and route each packet. If the destination options header contains only options for the final destination, it can be included in the fragmentable portion of the IPv6 packet.

2.1.2.5 AH/ESP header

The Authentication Header (AH) and Encapsulating Security Payload (ESP) header are required to support security (IPSec) for IP packets.

The AH provides data integrity and data authentication (verifies the node that sent the packet). They provide source node verification, verification that data was not modified in transit, and the ability to prevent nodes from retransmitting packets as valid data (anti-replay protection). This header provides no encryption on the data. The ESP header provides data confidentiality by encrypting data in addition to data integrity and data authentication. It can also be used in conjunction with the AH to provide additional packet verification.

If the AH is used with the ESP header, it should precede it. The ESP provides encryption for all data following the header. Any data or headers preceding the ESP header will not be encrypted.

IPSec supports two modes of operation, Tunnel and Transport mode. In Transport mode, the ESP/AH headers are inserted after the IP header and before the next layer protocol header. In Tunnel mode, ESP/AH headers are inserted after the initial IP header but before another encapsulated IP header.

3 Mobile Implementation Features and Limitations

3.1 AMSS IPv6 features

AMSS IPv6 is supported for the following wireless network technologies:

- cdma2000 (1X, 1xEV-DO, 1xEV-DO Rev A, etc.)
- UMTS (GPRS, WCDMA, HSDPA, etc.)

3.1.1 Supported features in AMSS IPv6 implementation

Table 3-1 lists the base features supported by the AMSS IPv6 implementation and associated limitations, if any. The features outlined here are described in detail throughout this chapter. These features are uniformly supported across all network technologies.

Table 3-1 Supported AMSS IPv6 features

Feature	RFC (if applicable)	Limitations
IPv6 protocol	RFC 2460, 2461, 2462	
Dual stack (IPv4 and IPv6) for embedded applications [S11]		
IPv6 extension headers	RFC 2460	No support for hop-by-hop and destination options (RFC 2460 Sections 4.3 and 4.4); see Sections 2.1.2.2 and 2.1.2.3 for more information
Neighbor discovery	RFC 2461	Neighbor discovery procedures are not supported for any WWAN interface. All supported WWAN technologies assign a unique prefix to the device during IPv6 address autoconfiguration and hence do not require any neighbor discovery procedures to be run on the wireless interface. Complete neighbor discovery support exists for tethered connectivity over RmNet.
DSS sockets API extension for IPv6	RFC 3493	
IPv6 fragmentation and reassembly	RFC 2460	

Feature	RFC (if applicable)	Limitations
ICMPv6 error messages	RFC 2463	See Section 3.3.
IPv6 loopback support	RFC 2460	
Utility functions for address conversion	RFC 2373	
Privacy extensions for stateless address autoconfiguration	RFC 3041	
IPv6 configuration parameters*		
DHCPv6 client	RFC 3315	Stateless configuration only; stateful address configuration is not supported over DHCPv6
IPv6 DNS support for querying A and AAAA records over IPv6 or IPv4	RFC 2874	SRV records are not supported. API support for querying CNAME records does not exist, however, if CNAME record is included in the reply, API support exists to return the record to the application

*For more information on supported IPv6 configuration parameters, see Chapter 4.

Table 3-2 lists the AMSS IPv6 features that are technology-dependent. For technology-specific information on these features, refer to corresponding technology-specific sections in this document. If a specific document is referenced, refer to it for more details.

Table 3-2 Technology-dependent IPv6 features in AMSS

Feature	RFC (if applicable)	Technology support			
		cdma2000	UMTS	eHRPD	LTE
IPv6 for external devices		Yes	Yes	Yes	Yes
Simultaneous support for embedded (IPv6 or IPv4) and external devices*		No	Yes	Yes	Yes
Header compression – ROHC [Q3]	RFC 3095	Yes	Yes	Yes	Yes
Header compression – IPHC	RFC 2507	No	Yes	No	Yes

*Laptops, PDAs, or external processors connected via serial buses, e.g., USB or data card

3.1.2 Feature limitations

Table 3-3 lists the IPv6-related features currently not supported by AMSS.

Table 3-3 Unsupported AMSS IPv6 features

Feature	RFC #	RFC section	Description
Path MTU discovery	RFC 1981, 2463		Discovers the minimum MTU supported along a path to the destination. AMSS IPv6 implementation limits IPv6 MTU to 1280 bytes. RFC 3316 and RFC 2460 allow for such restriction to mitigate the lack of Path MTU discovery support.
Any form of IPv6 tunneling			No support for automatic tunneling exists, including 6to4 or 4to6 (DS-lite) tunneling which are sometimes used as transition mechanisms for IPv6 deployment.

3.1.3 Implementation limitations

Table 3-4 lists current AMSS IPv6 implementation limitations in the AMSS IPv6 stack.

Table 3-4 AMSS IPv6 implementation limitations

AMSS IPv6 limitation	RFC #	RFC section	Description
ICMPv6 RAs (address configuration)	RFC 2461	4.2	At present, the mobile supports only one IPv6 prefix per interface. The first valid prefix received will become the interface's prefix. All subsequent prefixes received will be discarded as described in Section 3.5.1.
Routing header	RFC 2460	4.4	The AMSS IPv6 implementation processes the routing header in any incoming packet and discards the packet if the mobile node is not the next hop in the routing header. AMSS IPv6 implementation does not add routing header in any transmitted IPv6 packets.
MTU option in the RA	RFC 2461	4.6.4	The AMSS IPv6 implementation supports an MTU of 1280 or smaller. See Section 3.5.2 for more information.
Source link layer option in the RA	RFC 2461	4.6.1	This option is ignored while processing an RA.

3.2 IPv4/IPv6 dual-stack architecture

The AMSS data stack provides a dual-stack solution that supports IPv4 and IPv6. CDMA and UMTS technologies support both IPv4 and IPv6 and represent each protocol by separate interfaces. For more information on how IPv6 works in each technology, see Chapters 5 and 6.

The stack in Figure 3-1 utilizes one common IP input and transmit function. This allows layers above and below IP layer to be agnostic to the IP type. The layers above IP layer (transport and sockets) supports both IPv4 and IPv6 although they use IPv6 addressing in order to implement an IP agnostic architecture. For more information on setting up an IPv6 data session, see Chapter 4.

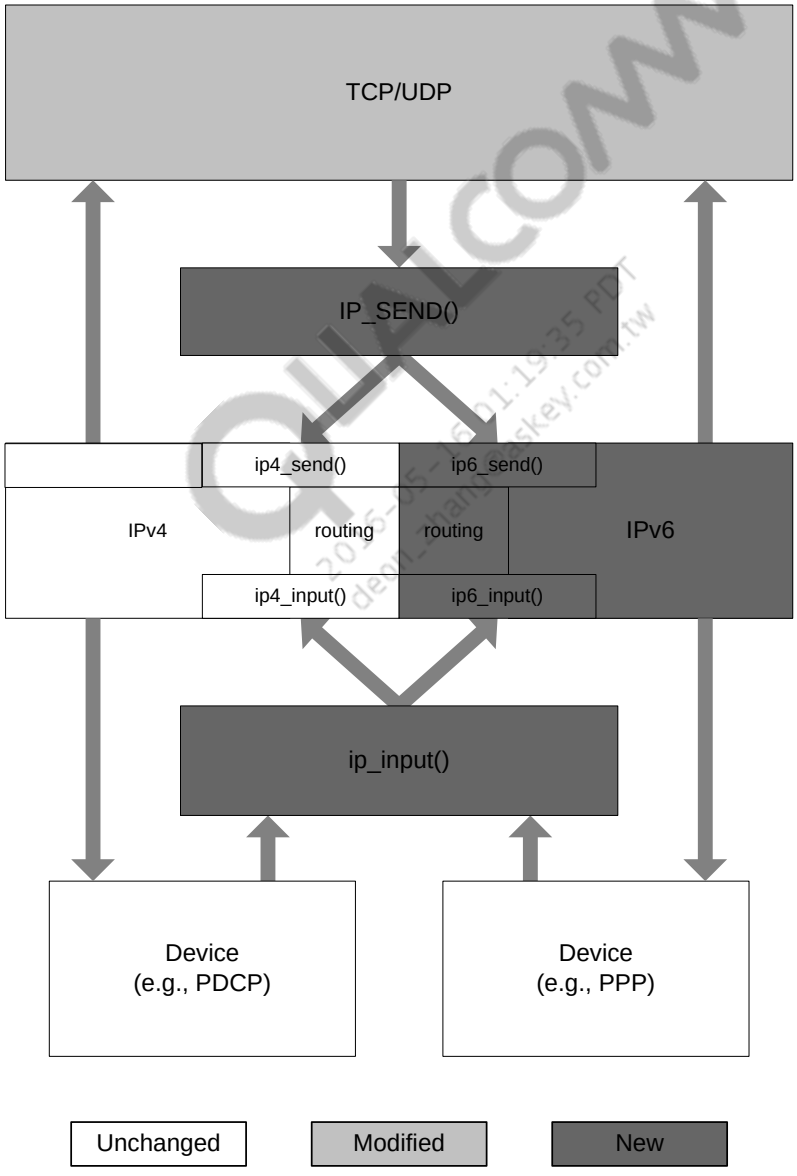

Figure 3-1 Dual IPv4/IPv6 stack structure

3.3 ICMPv6 messaging

The primary mechanism used for configuration and messaging in IPv6 is ICMPv6. RSs and RAs are the currently supported mechanisms for retrieving and maintaining IPv6 addresses on the mobile through the IPv6 autoconfiguration mechanism. In addition, the AMSS data stack provides ICMPv6 support for propagating informational and error messages back to applications. The following sections describe the format of these messages, as well as scenarios in which they are sent and received.

Figure 3-2 illustrates the basic ICMPv6 message format.

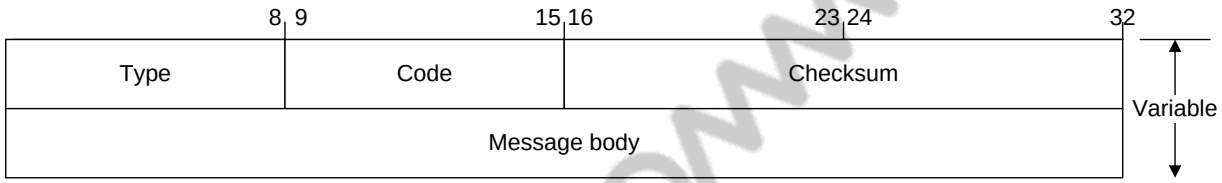

Figure 3-2 Basic ICMPv6 header format

3.3.1 ICMP message processing

Table 3-5 lists some of the ICMPv6 messages in the AMSS IPv6 implementation. Note that not all codes listed are supported. Messages not listed are unsupported.

Any actions taken on these messages are listed under the Action column. All interested applications will be notified of ICMPv6 error messages (Type 1-4) when they are received. Messages sent in response to an error condition are noted in Table 3-5 along with the cause.

All ICMPv6 informational/error messages are defined in [S6]. RA/RS messages are defined in [S4].

Table 3-5 ICMPv6 message support

ICMPv6 message	Type	Code	Code field value	Action
Destination unreachable	1	0	No Route to Destination	Notify application
		1	Communication with Destination Administratively Prohibited	Notify application
		2	Beyond Scope of Source Address	Notify application
		3	Address Unreachable	Notify application
		4	Port Unreachable	Notify application
Packet too big	2	0		Notify application
Time exceeded	3	0	Hop Limit Exceeded in Transit	Notify application
		1	Fragment Reassembly Time Exceeded	Notify application
Parameter problem	4	0	Erroneous Header Field	Notify application
		1	Unrecognized Next Header Type	Notify application
		2	Unrecognized IPv6 Option	Notify application
Echo request	128	0		Respond with an echo reply

ICMPv6 message	Type	Code	Code field value	Action
Echo reply	129	0		Silently discarded when received; sent in response to an echo request
Router solicitations	133	0		Silently discarded when received; sent to solicit for a router advertisement
Router advertisements	134	0		Process advertisement for prefix and MTU options

ICMPv6 error reporting

The AMSS IPv6 implementation supports application notification of ICMPv6 error messages received. This is accomplished in two ways:

- A socket option can be set and the error retrieved through a variable in the socket.
- The ICMPv6 error messages can be passed back as ancillary data through a receive call. Note that this is supported only for UDP.

For more details and usage of the AMSS APIs, see [Q2].

3.3.2 ICMPv6 error generation

Table 3-6 describes the circumstances under which the mobile will send an ICMPv6 error message.

Table 3-6 ICMPv6 error generation scenarios

ICMPv6 message	Type	Code	Code field value	Description of when this message is sent from the mobile
Destination unreachable	1	4	Port Unreachable	UDP packet is sent to mobile with no server running for the port
Time exceeded	3	1	Fragment Reassembly Time Exceeded	Fragment reassembly timer expires because it is not able to reassemble a fragmented IPv6 packet in time
Parameter problem	4	0	Erroneous Header Field	■ When an IPv6 packet is sent to the mobile with an unrecognized protocol ■ Erroneous header field (pointing to routing header type) is sent when an IPv6 packet reaches the mobile containing a routing header with routing type field and segments left field both nonzero. ■ Erroneous header field (pointing to segments left field) is sent when an IPv6 packet reaches the mobile with a type zero routing header with segments left field nonzero. ■ Erroneous header field (pointing to payload length field of a fragment packet) is sent if the mobile receives a fragment such that the length of the fragment is not a multiple of 8 octets and the M flag of that fragment is 1. ■ Erroneous header field (pointing to fragment offset field of a fragment packet) is sent if the mobile receives a fragment whose length and offset are such that the payload length of the packet reassembled from that fragment would exceed 65535 octets.

ICMPv6 message	Type	Code	Code field value	Description of when this message is sent from the mobile
		1	Unrecognized Next Header Type	IPv6 packet is sent to mobile with a routing header containing unrecognized next header

3.4 RSs

RSs are sent by the mobile to solicit the router, e.g., PDSN/GGSN, for an IPv6 prefix (RA). The two scenarios in which the MS will send these are described as follows.

On initial call setup, if an RA is not immediately received, the mobile waits a number of seconds defined by IP6_SM_DEFAULT_INIT_SOL_DELAY to solicit for one. The mobile will then send an RS n times, defined by the configuration parameter IP6_SM_DEFAULT_MAX_SOL_ATTEMPTS, until a valid one is received. Each RS is sent at an interval defined by IP6_SM_DEFAULT_SOL_INTERVAL. If one is not received, the MS fails and tears down the IPv6 session. For more information on these configuration parameters, see Chapter 4.

If the MS succeeds in setting up an IPv6 session, the mobile must resolicit for a new address in the following two cases:

1. Expiration of the preferred lifetime – At the time when the prefix transitions to deprecated state; at this point the mobile resolicits for an RA. If one is received and the prefix is the same, the lifetimes will be updated and the prefix will be moved back to valid state. Currently, we do not support different prefixes from the same router (non-handoff).
2. Pre-RA-expiry RS timer – Time before which the router lifetime, from the original RA, expires and the mobile should begin resoliciting for an RA. If the value of the pre-RA-expiry RS timer is greater than 50% of the router lifetime, the MS starts resolicitation at 50% of the router lifetime. Otherwise, it uses the value set in the pre-RA-expiry configuration. If resolicitations fail to get an RA, the mobile continues to resolicit, subject to geometric backoff, until the prefix expires.

See Chapter 6 on how to configure the pre-RA-expiry RS timer. See Section 3.4.1 for example call flows of the two above scenarios.

Resolicitation is implemented as follows. The first RS is sent at the time when the MS starts the resolicitation procedure. The second RS is sent after the IP6_SM_DEFAULT_RESOL_ INTERVAL from the first RS. Subsequent RSs are subject to geometric backoff. The total number of resolicitation RS messages the MS sends is determined by IP6_SM_DEFAULT_ MAX_RESOL_ATTEMPTS parameter.

Table 3-7 lists the RS format and values the mobile sends. This includes both the IPv6 and ICMPv6 header and values.

Table 3-7 RS message

Protocol	IP fields	Value	Length (bits)	Description
IPv6 header	Version	6	4	IPv6 version number
	Traffic class	0	8	Type of traffic
	Flow label	0x00000	20	Traffic classification (QoS related)

Protocol	IP fields	Value	Length (bits)	Description
	Payload length	8	16	Length of the data (ICMPv6 header in this case)
	Next header	58	8	The next protocol header, in this case ICMPv6
	Hop limit	255	8	Maximum number of hops until IP datagram is discarded
	Source address	::0 or FE80::MS-IID	128	Unspecified address of the mobile or If mobile has already configured an address and is resoliciting, it will use its link-local address
	Destination address	FF02::2	128	IPv6 link-local all routers multicast address
ICMPv6 header	Type	133	8	RS
	Code	0	8	
	Checksum		16	
	Reserved	0	32	Does not need to be specified

3.4.1 RS call flows

The call flow in Figure 3-3 illustrates the mobile resoliciting for an RA after the prefix's preferred lifetime expires. After the prefix moves to the deprecated state, no new sockets can make connections using that prefix. Only connections that were made prior to the prefix's expiration are allowed to send/receive data. Once the prefix lifetimes are renewed, any new sockets can use this prefix to transfer data.

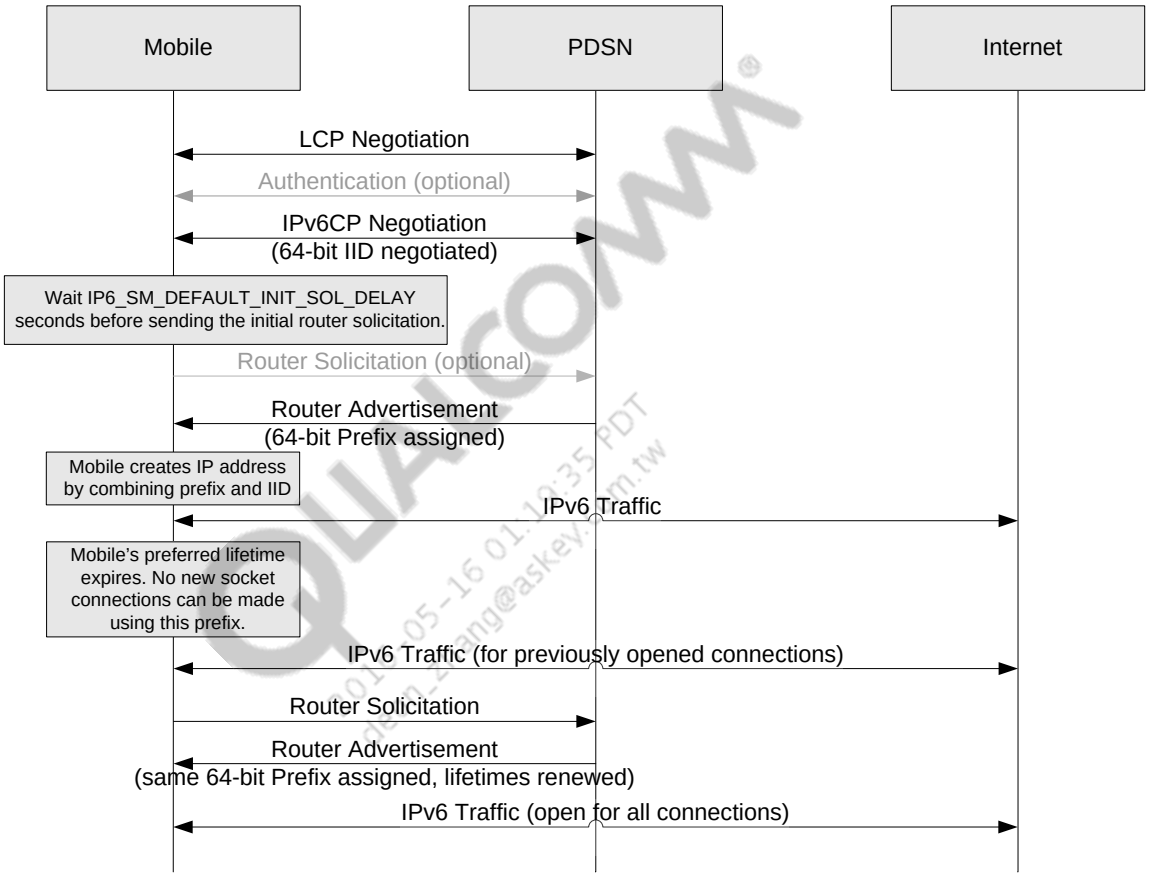

Figure 3-3 Router solicitation sent after expiration of the preferred lifetime

The call flow in Figure 3-4 illustrates the mobile resoliciting for an RA after the prefix's preferred lifetime expires as no RA is received. In that case, only existing socket connections can continue to send data using this prefix. When the pre-RA-expiry RS timer expires, the mobile again solicits for a RA. When it receives a new RA and a valid prefix is configured, any socket is allowed to make connections and transfer data.

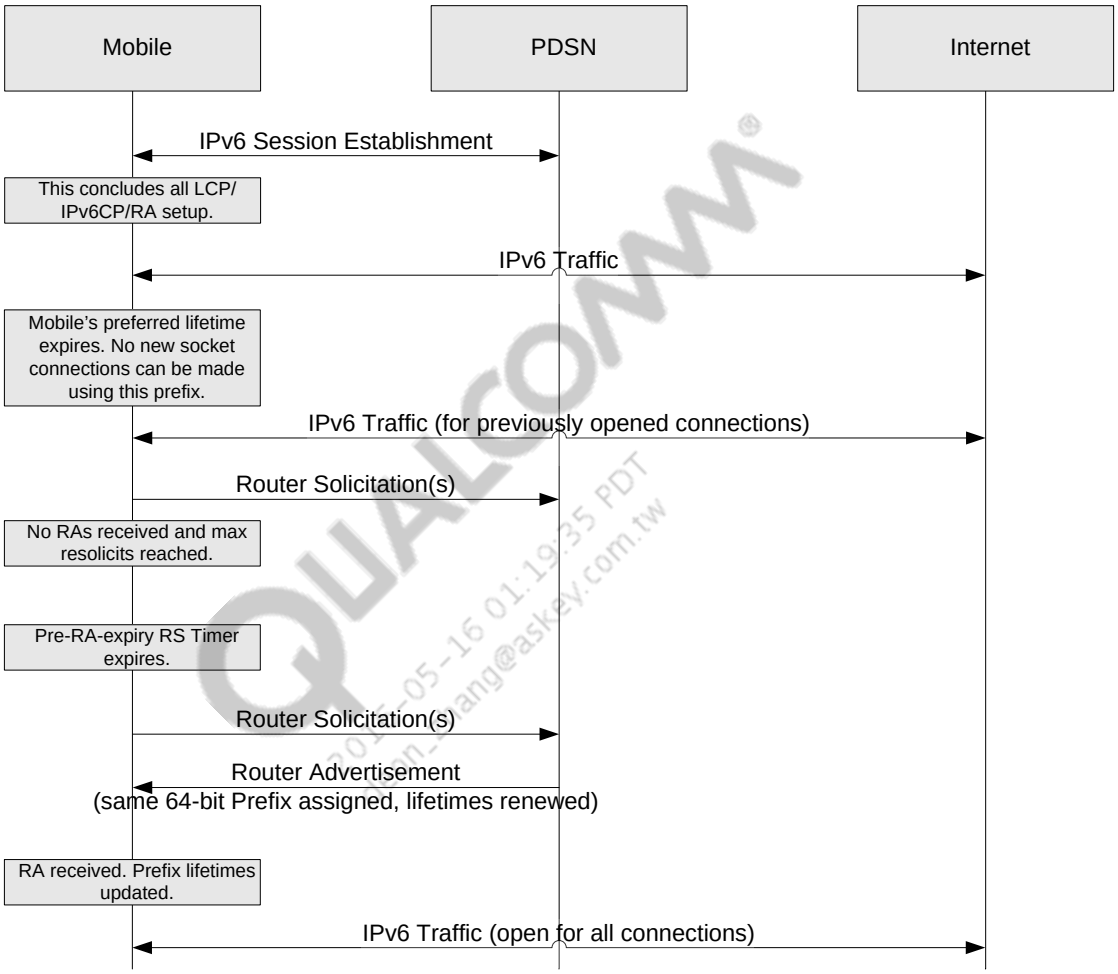

Figure 3-4 Router solicitation sent after expiration of the RA-expiry RS timer

The call flow in Figure 3-5 illustrates the mobile soliciting for an RA on initial IPv6 session setup. When the mobile does not receive an RA, it solicits for an RA the number of times specified by the mobile configuration parameters mentioned in Section 3.4. If it does not receive one, the session is terminated.

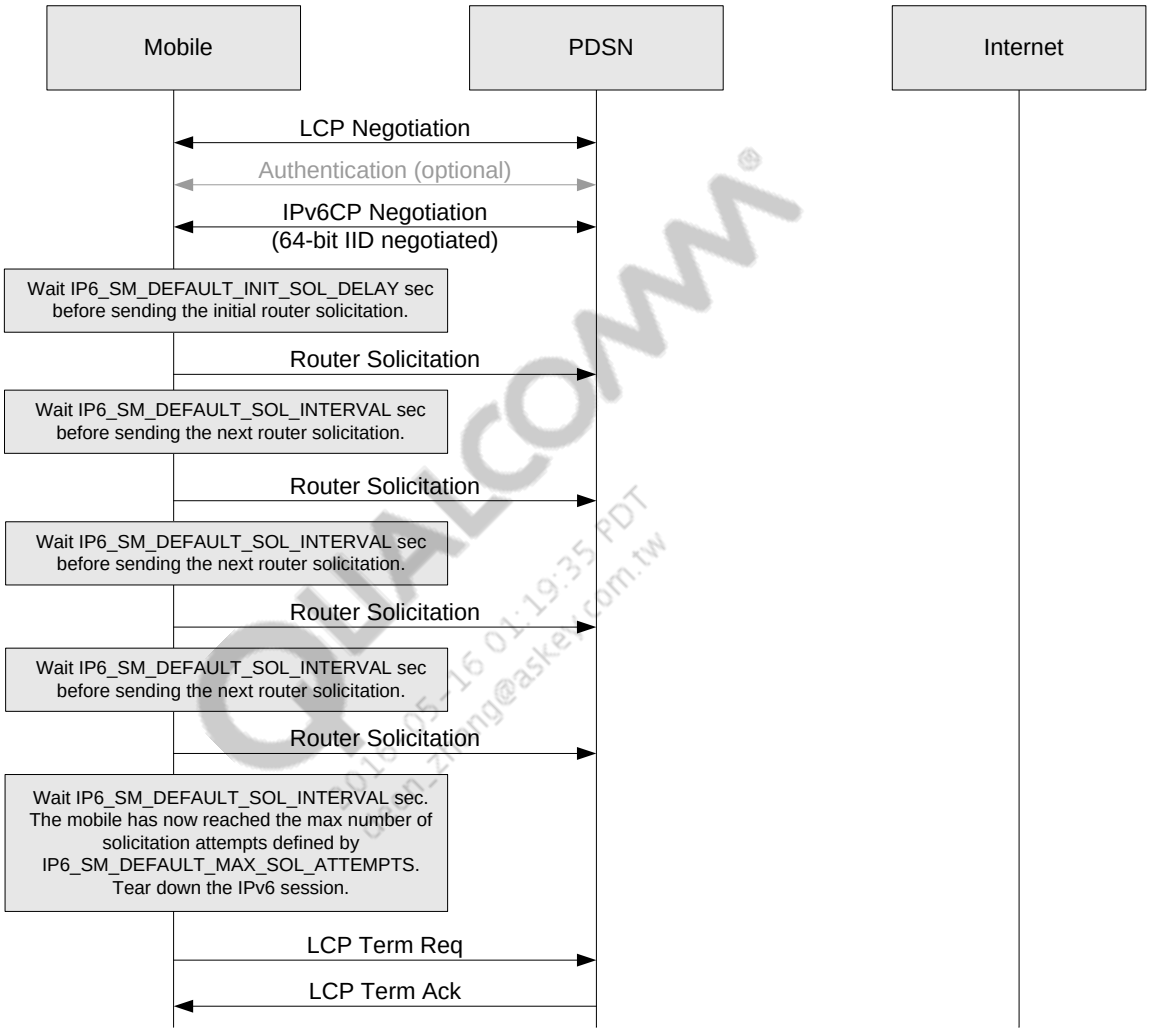

Figure 3-5 Initial solicitation retries occur when no initial router advertisement is received

The call flow in Figure 3-6 illustrates the mobile soliciting for an RA after the preferred lifetime has expired. After it resolicits, it does not receive an RA. When the Pre-RA-expiry RS timer expires, it begins a geometric resolicitation mechanism. The mobile continues to resolicit following a geometric backoff until the RA lifetime expires and the prefix is deleted. The mobile then terminates the IPv6 session.

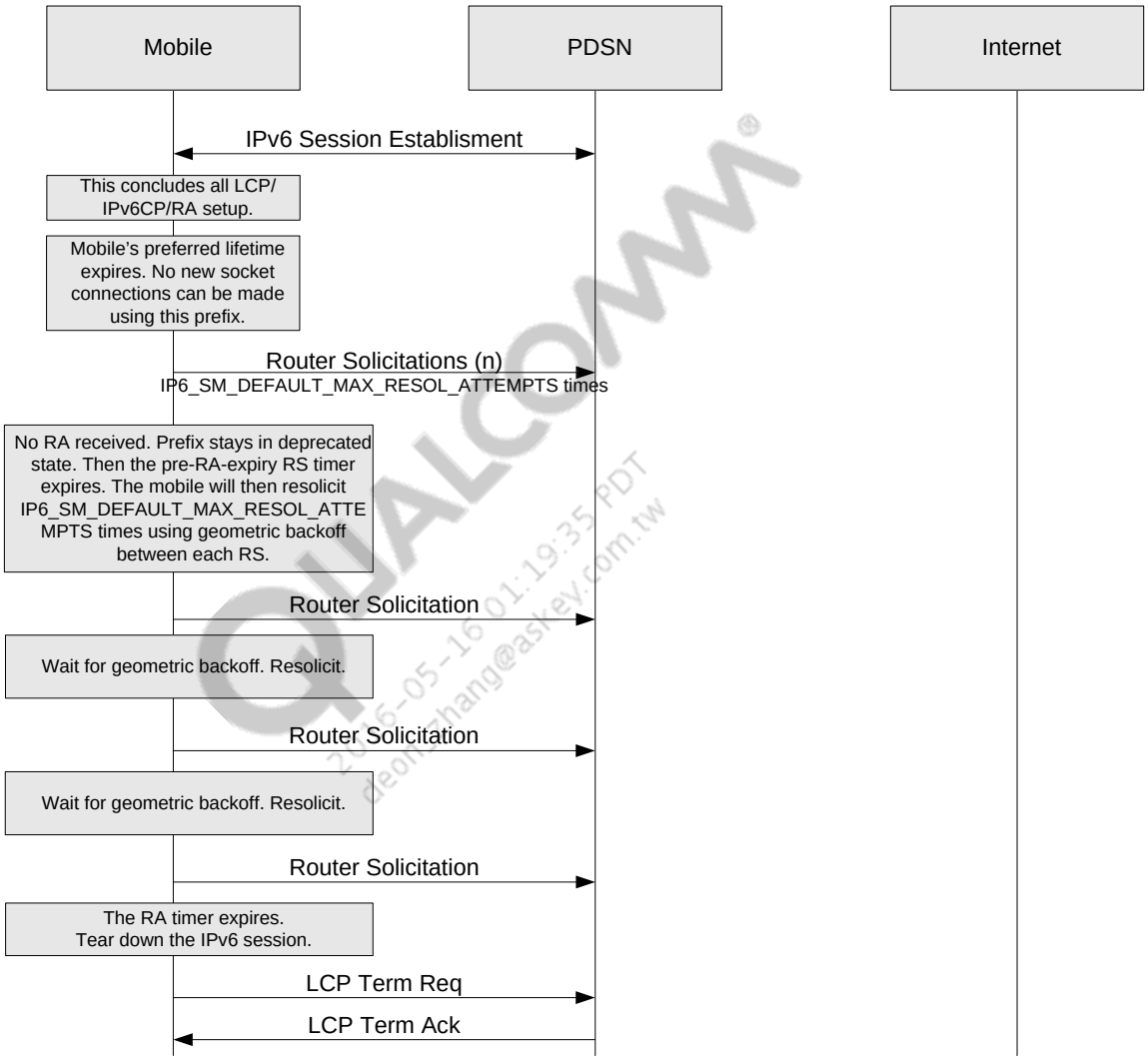

Figure 3-6 Resolicitations occur after the prefix expires

3.5 RAs

RAs are sent by the router to advertise IPv6 prefixes, one of which is used by the mobile. They are sent at periodic intervals preset by network administrators as well as in response to an RS sent by the mobile. The mobile does not support sending RAs. RAs can have multiple options contained within. The only options supported by the AMSS implementation are the MTU and prefix information options. Any other options received are silently discarded. The MTU is the maximum packet size supported over the link without having to use fragmentation. The prefix information options are processed as follows.

3.5.1 Prefix information option

Prefix options are parsed in the order in which they are listed in the RA. Currently, the mobile will only process the first prefix information option. All remaining prefix information options are ignored. All prefix options are validated prior to adding them successfully to an interface. The managed flag (M) in the RA header must be 0. The on link flag (L) and the autonomous flag (A) must be set to 0 and 1, respectively, in the prefix information option. If any of these flags is incorrectly set, the mobile ignores the prefix.

The prefix information option in the RA has two lifetimes associated with the prefix, the preferred lifetime and the valid lifetime. If either of the lifetimes specified in the prefix information option is zero, the RA is dropped.

- Preferred lifetime – Time for which the prefix state is considered preferred by the mobile. At its expiry, the mobile transitions the prefix state to deprecated and applications are notified. In the deprecated state, only existing connections (e.g., TCP) continue to be serviced. No new sockets are allowed to initiate data transfer by the mobile until the timer is refreshed.
- Valid lifetime – Time for which the prefix is valid. At the expiry of this time, the IPv6 session is brought down. This time is greater in value than the preferred lifetime.

Additionally, the RA contains router lifetime as one of its fields. At expiration of the router lifetime, assuming no new RA is received thus renewing the lifetime, the IPv6 session is brought down.

3.5.2 MTU option

The default MTU used by the mobile is 1280 bytes. Only an MTU value of 1280 or less is currently supported. If an MTU option is present in the RA and has an MTU value larger than 1280, it is ignored. If an MTU option has a value smaller than 1280, the MTU is reduced to that value.

Table 3-8 RA message with prefix option

Protocol	IP field	Value	Length (bits)	Description
IPv6 header	Version	6	4	IPv6 version number
	Traffic class	0	8	Type of traffic
	Flow label	0x00000	20	Traffic classification (QoS-related)
	Payload length	48	16	Length of the data
	Next header	58	8	Next protocol header, ICMPv6

Protocol	IP field	Value	Length (bits)	Description
	Hop limit	255	8	Maximum number of hops until IPv6 datagram is discarded
	Source address	FE80::ROUTER-IID	128	Sent from the IPv6 link-local address of the router
	Destination address	FE80::MS-IID or FF02:1	128	First address is expected when sending to the mobile in response to an RS; the second—the IPv6 link-local all nodes multicast address—is used for unsolicited advertisements
ICMPv6 RA header	Type	134	8	RA
	Code	0	8	
	Checksum		16	
	Curr hop limit	255	8	
	Config flags	0	8	
	Router lifetime	65535	16	Length of time that the router is valid as a default router
	Reachable time	0	32	Amount of time a node can consider a neighboring node reachable after receiving a reachability confirmation
	Retransmit time	0	32	Amount of time between retransmissions of neighbor solicitations
ICMPv6 prefix option (one or more instances)	Type	3	8	Prefix option
	Length	4	8	Length value in octets
	Prefix length		8	Must be 64 bits for CDMA/UMTS technologies
	L	0	1	On-link flag
	A	1	1	Autonomous flag
	Reserved1	0	6	
	Valid lifetime		32	Must be nonzero
	Preferred lifetime		32	Must be nonzero and smaller than the value of valid lifetime
	Reserved2	0	32	
	Prefix		128	
MTU option	Type	5	8	MTU option
	Length	1	8	Length value In octets
	Reserved	0	16	Unused
	MTU		32	MTU value (only 1280 or less supported)

3.6 Privacy extensions for stateless address autoconfiguration

RFC 3041 provides a mechanism for generating private IPv6 addresses that are more difficult to track, the intention being that these private addresses will be changed consistently over some shorter periodic basis than the fully qualified “public” address established at the initial connection.

The AMSS IPv6 implementation of RFC 3041 supports:

- The ability to enable/disable support for RFC 3041 at runtime.
- A set of configurable parameters for IPv6 privacy extensions.
- The ability to generate new privacy addresses upon application request.
- The ability to maintain at least one validated available address so applications can switch to their new addresses with minimal delay. By generating an address prior to application requests, lengthy setup times can be prevented for technologies that require additional address verification, e.g., broadcast interfaces using DAD.
- The ability for applications to designate whether they want addresses solely for their use or to share a system-wide private IPv6 address.

3.6.1 Privacy address generation

There are two types of private IPv6 addresses supported in the AMSS IPv6 implementation, private shareable and private unique. These privacy addresses are created solely upon application request. No valid lifetimes are currently supported for privacy addresses to prevent termination of an address during critical applications, such as VoIP. They are deleted as stated in the following subsections.

3.6.1.1 Private shareable

A private shareable IPv6 address adheres to the rules of RFC 3041 and is shareable among any number of sockets. There can be only one valid private shareable address per interface.

The deprecation lifetime timer for a shareable privacy address will be started when the first socket binds to this address. Subsequent sockets that bind to this address will not affect the timer. The expiration of the timer will cause a deprecation event to be generated to all applications using the address. It is suggested that applications then request a new shareable address. Once the address is deprecated and all applications have unbound from this address, or the IPv6 session is terminated, it is deleted.

3.6.1.2 Private unique

A private unique IPv6 address adheres to the rules of RFC 3041 but is unique to a particular application. This address is owned by one particular application and cannot be shared amongst applications. There can be up to (MAX_IPV6_PREFIXES * (MAX_IPV6_IIDS -1)) private unique addresses per interface.

The first bind to this address causes the deprecation lifetime timer to begin. Once all the sockets unbind, or the IPv6 session is terminated, the address is deleted. Expiration of the preferred lifetime timer causes a deprecation event to be generated to the application using the address. It is also suggested that an application request a new private unique address at this point.

3.7 Mobile station IPv6 programming interface

To establish an IPv6 session, an application must first set up its network policy to request an IPv6-enabled interface. The desired interface can be assigned by setting the family field in the application's network policy to the IP address family for the interface to be brought up. The valid values for this field are:

```
AF_INET  /* IP version 4 */
AF_INET6 /* IP version 6 */
AF_UNSPEC /* Any IP address family */
```

If an application *requires* IPv6, it must explicitly ask for it by requesting an interface with the family AF_INET6. If the application is IP-agnostic, and can use either v4 or v6, then it may specify the family AF_UNSPEC. The specifics of how the mobile determines which interface to use when specifying AF_UNSPEC are covered in Section 5.3, as the behavior differs between technologies.

To transit and receive data on an IPv6 interface, an application must open an IPv6 socket. An IPv6 socket can be used to transmit either IPv4 or IPv6 data, assuming the respective IP interfaces are up and sessions negotiated. An IPv4 socket cannot, however, be used for sending and receiving IPv6 data. For more information on IPv6 and any AMSS APIs, see [Q2].

3.7.1 Maximum transfer unit

The default MTU used by the mobile is 1280 bytes. This meets the IPv6 minimum required supported MTU of 1280 bytes, and, hence, path MTU is not necessary.

3.7.2 IPv6 loopback support

Loopback over IPv6 is supported in the AMSS IPv6 implementation. Sending IPv6 data to either the IPv6 loopback address, ::1, or any IPv6 address assigned to a local IPv6 interface causes data to be routed back over the IPv6 loopback interface.

3.7.3 Usage

Header files

AMSS IPv6 sockets APIs are declared in dssocket.h. Address conversion routines are declared in dssdns.h.

Featurization

Use FEATURE_DATA_PS_IPV6 to enable the IPv6 protocol and interfaces.

Compile time constants

The following compile time constants are not configurable:

```
MAX_IPV6_PREFIXES 1    /* Maximum supported IPv6 prefixes on the mobile */
MAX_IPV6_IIDS 5        /* Maximum supported IPv6 interface identifiers  */
```

4 Mobile Station IPv6 Configuration

Table 4-1 describes the configurable IPv6 parameters that are stored in nonvolatile RAM and their default values. Note that all of the following settings can be configured only if the IPv6 feature FEATURE_DATA_PS_IPv6 has been compiled into the build.

Table 4-1 Configuration settings

NV setting	Description	Default value(s)	Range	Technology support			
				cdma2000	UMTS	eHRPD	LTE
IPv6 enabled	Determines whether the IPv6 is supported; if False, none of the following settings are meaningful and they are ignored.	True	True/ False	Yes	Yes	Yes	Yes
PDSN as proxy DNS	When enabled, the mobile forwards all DNS requests to the PDSN. The PDSN forwards requests to the appropriate DNS server. This parameter is meaningful only if the primary and secondary DNS server addresses are not available.	True	True/ False	Yes	No	No	No
Primary and secondary DNS servers[1]	Primary and Secondary IPv6 DNS servers can be configured here. All servers should be in the standard IPv6 address format of eight groups of two hex octets each separated by colons.	N/A	Must be valid IPv6 addresses	Yes	No	Yes	No

[1]Primary and secondary IPv6 DNS servers – Each of these servers must be configured in the following format: FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF. This option is not supported by UMTS technologies.

NV	Description	Default	Range	Technology support			
RS settings	RSs are used to request an RA from the router before an unsolicited advertisement arrives; each parameter is described in detail below	Initial solicitation delay – 300 ms	0 to 2^{16} -1	Yes	Yes	Yes	Yes
		Solicitation interval – 1000 ms					
		Resolicitation interval – 2000 ms					
		Max solicitation attempts – 3					
		Max resolicitation attempts – 5					
		Pre-RA expiry resolicitation time – 0					
Unspecified address family behavior	Defines which IP family will be established for applications specifying AF_UNSPEC; for details, refer to the technology-specific sections	CDMA – IPv6 only UMTS – IPv4 only	IPv4 only IPv6 only	Yes	Yes	Yes	Yes
IID type IID user value	Interface identifier, which can be preprovisioned or randomly generated for each IPv6 session; for details, refer to the technology-specific sections	Random	Type – Random/ pre-provisioned User value – (0 to 2^{64}-1)	Yes	No	No	No

NV	Description	Default	Range	Technology support			
IPv6CP	IP control protocol used to configure and negotiate the PPP session for IPv6	Config request tries – 20 Config request timeout – 1000 ms Config NAK tries – 3 Term request tries – 3 Term Request timeout – 1000 ms Compression setting – ignore	0 to (2^{16} -1)	Yes	No	No	No
IPv6CP remote initiate accept	Allows the mobile to either ignore or bring up a PDSN-initiated NCP config request	Ignore	Ignore/ Bring up (B/U)	Yes	No	No	No
IPv6 privacy extensions enabled	Determines whether IPv6 privacy extensions is supported; if False, none of the privacy extensions lifetime settings are meaningful and they are ignored	True	True/ False	Yes	Yes	Yes	Yes
IPv6 privacy extensions lifetimes	Used to configure the privacy address lifetimes; only the preferred lifetime is currently supported	1 day	0 to (2^{32} -1)	Yes	Yes	Yes	Yes

4.1 RSs

RSs are used to request new RAs. If the mobile's address is going to expire, it can request a new one before becoming invalid. The following settings allow the mobile to request a new address earlier/later and more/less often than the default settings:

- Initial solicitation delay – Time mobile waits after the IID has been negotiated before sending an RS in an attempt to receive an RA
- Solicitation interval – Amount of time the mobile waits before sending a subsequent RS after a previous one

- Resolicitation interval – Amount of time between RSs sent while resoliciting for a *new* RA. This interval applies only after the mobile has previously received one valid RA and is soliciting for a new one to renew the lifetimes of the current prefix or retrieve a non-deprecated prefix.
- Max solicitation attempts – Number of solicitation attempts to make for initial IPv6 session setup, when an RA is not received in response before giving up IPv6 autoconfiguration
- Max resolicitation attempts – Number of solicitation attempts to make to resolicit for a *new* RA
- Pre-RA expiry resolicitation time – Amount of time before the current RA expires to begin resolicitations

See Section 3.4 and Section 3.5 for more information on RSs and RAs.

5 cdma2000-Specific Support

The following section describes IPv6 behavior specific to cdma2000 technology [S11].

5.1 Simple IP call flows

Setting up an IPv6 session is similar to establishing an IPv4 one. In cdma2000 systems, both IPv4 and IPv6 require that PPP be successfully negotiated to receive a valid IP address and DNS addresses. For successful simple IPv4 address configuration, LCP and the IP Control Protocol (IPCP) must first be established. For IPv6, IPv6CP replaces IPCP, and in addition, a valid RA must be received. Figure 5-1 is an example of a simple IPv4 session setup.

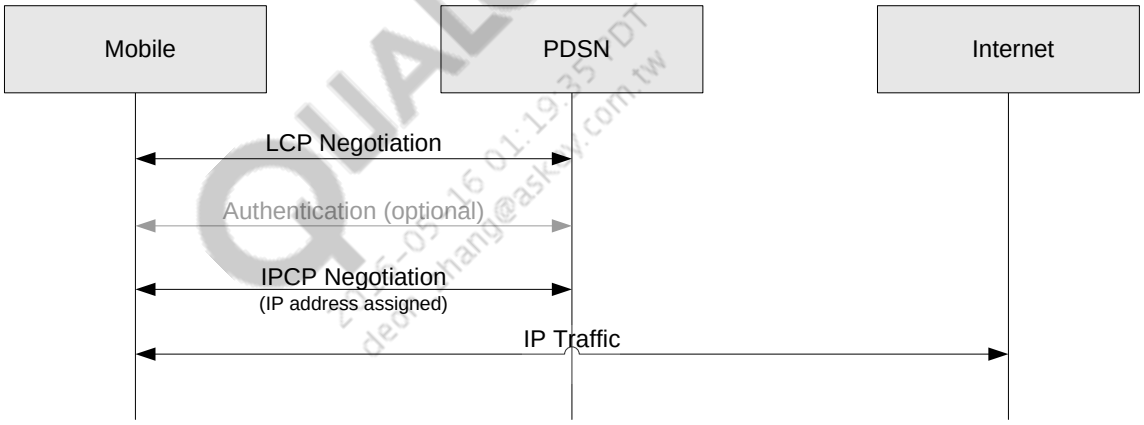

Figure 5-1 cdma2000 simple IPv4 negotiation

Figure 5-2 describes IPv6 session negotiation (assuming no IPv6 session is previously set up). IPv6 uses the IPv6 Control Protocol (IPv6CP) to negotiate its IID. In addition to LCP and IPv6CP, the mobile must also receive a valid RA with a 64-bit prefix to form its IPv6 address. In IPv6, the address is obtained by appending the 64-bit IID to the 64-bit prefix received in the RA.

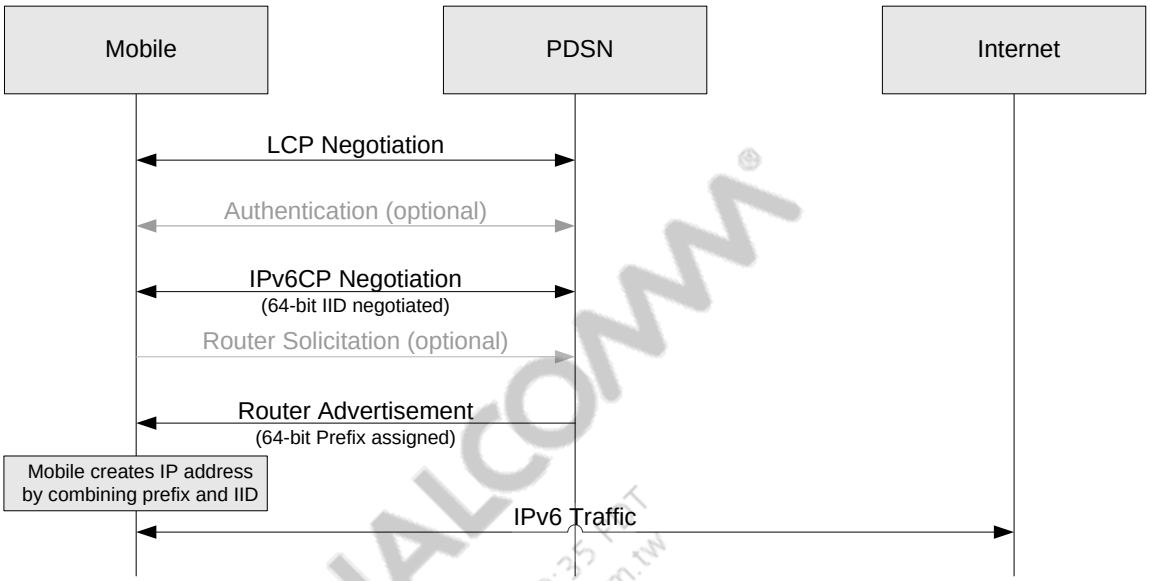

Figure 5-2 cdma2000 simple IPv6 negotiation

5.2 Technology-specific configuration parameters

The following IPv6 configuration parameters are supported only for cdma2000:

- IID – The mobile's interface identifier, or last 64 bits of the fully qualified IPv6 address. This can be selected to be random, in which case a 64-bit IID is generated by the mobile at power-up, or it can be prespecified during provisioning of the mobile. Neither of these IIDs are guaranteed to be used, of course, if there is a collision between the two negotiated IIDs during IPv6CP negotiation (RFC 2462). If there is a collision, both the mobile and PDSN must generate another random IID.
- IPv6CP settings – These settings are identical to those used for IPCP.
- Remote-initiated IPv6CP – If a PDSN has the IPv6CP NCP set to Active mode, it may attempt to initiate an IPv6 session on its own. If the mobile has the remote-initiated IPv6CP setting to Ignore, it simply drops the configuration requests. If set to Bring Up, it continues negotiation of the IPv6CP protocol and sets up an IPv6 session. There is an identical setting available for remote-initiated IPCP sessions as well.

5.3 IP-agnostic applications

For cdma2000 interfaces, if an application specifies AF_UNSPEC as the IP family type in the policy, the IP family configured in Unspecified Address Family Behavior is used. If this configuration parameter is not configured, then the IP family will default to AF_INET6 (implies IPv6).

6 UMTS-Specific Support

This chapter describes IPv6 behavior specific to UMTS technology [S13].

UMTS supports only one IP type per PDP context. Therefore, only one type of IP traffic (IPv4 or IPv6) can be exchanged over the same interface and physical link at any given point. UMTS technology can support one or more IP interfaces and each can be of type IPv4 or IPv6.

In UMTS, applications sharing an interface must have matching policies, including the PDP profile number. Applications requesting different PDP profile numbers are bound to different interfaces and, hence, tied to different GGSNs.

6.1 IPv6 call setup

AMSS IPv6 implementation supports only PDP-IP call types. To set up a PDP-IP data session, IPv6 stateless autoconfiguration process is performed through 3GPP-specified signaling rather than PPP and IPv6CP (Figure 6-1). The IPv6CP IID option is sent through signaling from the GGSN to the SGSN, which relays the option back to the mobile. The assignment of the IID through the GGSN eliminates the need for the mobile to perform DAD. Instead, the GGSN picks both the mobile and gateway IID, ensuring that both are unique for the prefix assigned. The IPv6 DNS addresses are also configured through the context activation signaling, eliminating the need for preprovisioning of DNS addresses.

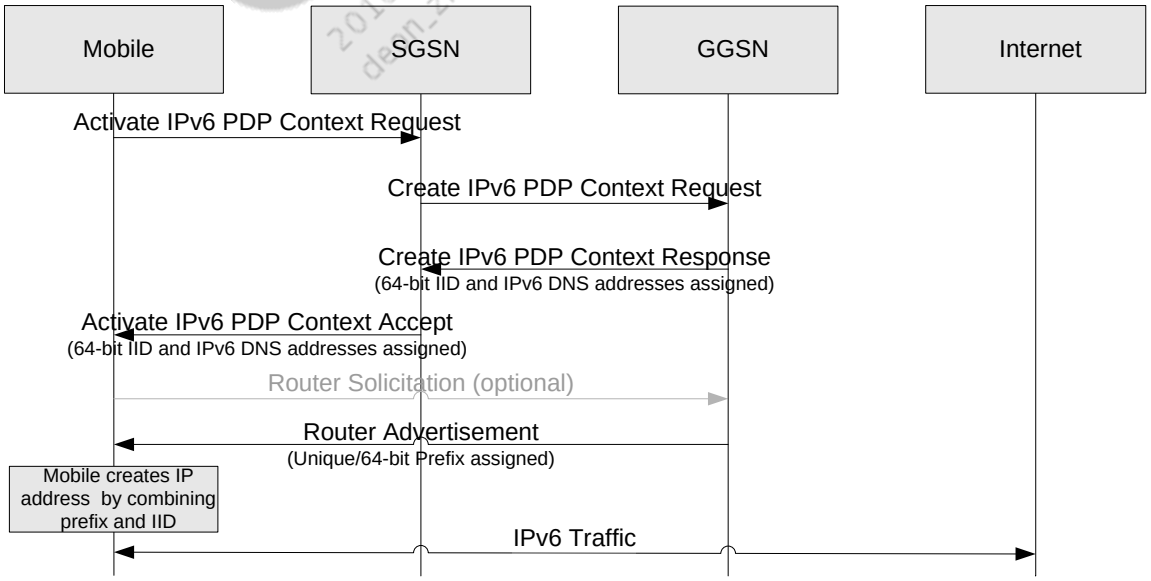

Figure 6-1 UMTS IPv6 session setup

6.2 IP-agnostic applications

For UMTS interfaces, if an application specifies AF_UNSPEC as the IP family type in the policy, the IP family is automatically mapped to AF_INET for IPv4. IP family AF_UNSPEC should be specified only if no specific profile is specified, in which case a default profile (profile 0) is automatically selected. An application should not specify AF_UNSPEC and an explicit profile number, as the application is expected to know the IP type of the profile and should specify the same IP type in the policy.

7 eHRPD-Specific Support

This chapter describes IPv6 behavior specific to eHRPD technology [S14].

eHRPD can support both IPv4 and IPv6 over the same PDN connection. However, IPv4 and IPv6 traffic going over the same PDN is managed via separate interfaces, one for each IP type. eHRPD can support multiple PDN connections at a time, each of which can support IPv6. eHRPD PDN connection setup requires LCP and authentication phase of PPP to be successfully negotiated. The network control protocol to establish the PDN connection and IPv6 address configuration is vendor-specific (3GPP2 VSNCP). All of the active PDN connections use the PPP link for the default data flow. Packets from different PDN connections are differentiated based on PDN-ID, which is negotiated during the PDN attach (VSNCP) procedures.

AMSS uses eHRPD profiles to aggregate the parameters that identify a particular PDN connection. An application can specify that an IPv6-enabled eHRPD profile bring up or bind to a PDN connection. Applications requesting different eHRPD profile numbers are bound to different interfaces and, hence, tied to different PDN connections.

7.1 IPv6 call setup

Figure 7-1 describes high-level call flow for IPv6 call setup in eHRPD. eHRPD requires stateless address autoconfiguration for IPv6 session setup. The PDN attach procedures help in negotiation of the IPv6 IID for the IPv6 address configuration. Since the network allocates and assigns the IPv6 IID to the MS, it eliminates the need for IID type configuration items. Besides that, it also eliminates the need of performing any DAD, as the network ensures a unique IID allocation for the assigned prefix. The IPv6 prefix is obtained via Router Advertisement (RA) after the VSNCP PDN attach procedure is complete. Since the network assigns a globally unique IPv6 prefix to the MS, it eliminates the need for any neighbor discovery over the air interface. The following procedure corresponds to the numbered steps in Figure 7-1.

1. eHRPD requires the LCP phase of PPP negotiations to be complete to negotiate link-level parameters and authentication protocol.
2. The MS authenticates with the network before any PDN connection can be established.
3. The MS initiates the PDN connection attach procedures by sending a VSNCP configure request. This message includes information about the APN, the IP network capability of the mobile and address allocation request, DNS server address allocation request, along with other options.
4. The VSNCP configure request from the MS triggers the HSGW to initiate PMIPv6 procedures to request IPv6 address allocation and tunnel establishment with the PDN-GW.

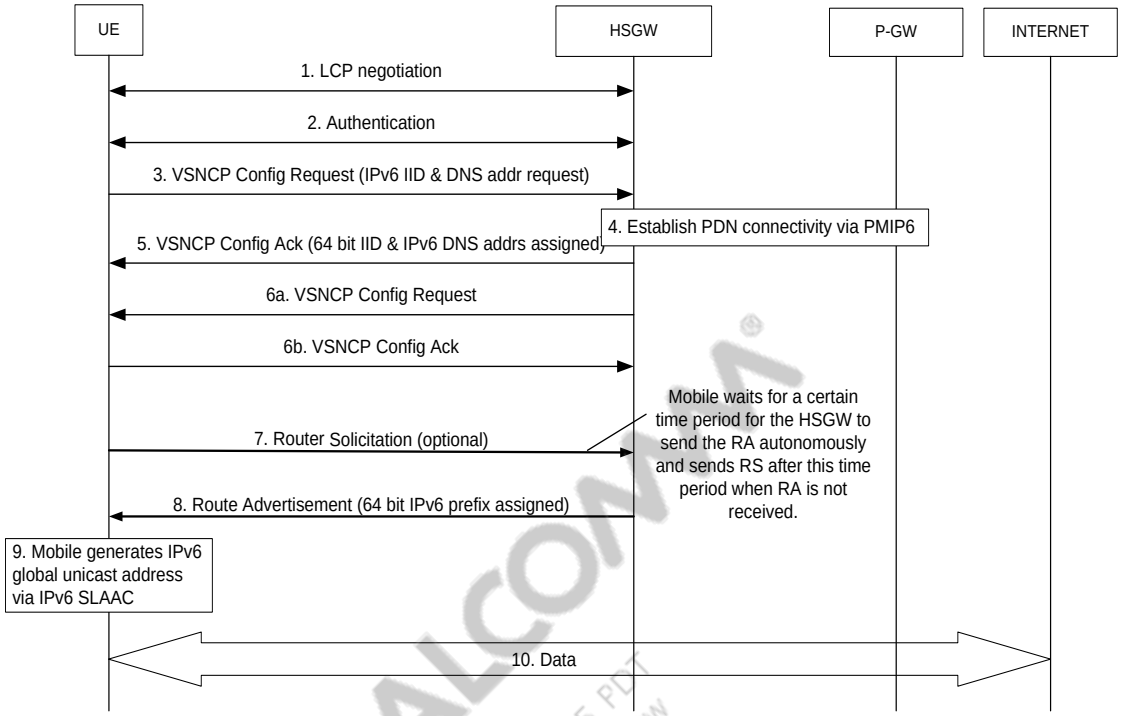

Figure 7-1 eHRPD IPv6 session setup

5. After the PMIPv6 procedures are complete, the HSGW has enough information to complete the IPv6 session setup with the MS. It responds to the configure request from the MS with a VSNCP configure Ack, which includes the allocated IPv6 IID and the IPv6 DNS server addresses for the MS.
6. Steps 6a and 6b complete the symmetrical VSNCP negotiation procedures.
7. Instead of sending a Router Solicitation (RS) immediately after the completion of VSNCP negotiation, the MS waits for a certain time period for the HSGW to send an unsolicited Router Advertisement. If the RA does not arrive within the specified timeframe, the MS sends RS to the HSGW.
8. The HSGW sends an RA to the MS including the unique IPv6 prefix assigned by the PDN-GW.
9. The MS generates a global unicast IPv6 address by combining the allocated IPv6 prefix and IID via the stateless address autoconfiguration procedures.
10. The MS can send/receive data using the IPv6 address configured via the above steps.

7.2 IP-agnostic applications

For eHRPD interfaces, if an application specifies AF_UNSPEC as the IP family type in the policy, the IP family is mapped to the IP family supported by the eHRPD profile specified by the application. If the application specifies AF_UNSPEC as IP family but does not specify any eHRPD profile, then the IP family is mapped to the one supported by the default eHRPD profile.

AMSS does not currently support dual-IP bearer PDN connections over eHRPD.

8 LTE-Specific Support

This chapter describes IPv6 behavior specific to LTE technology.

LTE technology can support both IPv4 and IPv6 over the same PDN connection. However, IPv4 and IPv6 traffic going over the same PDN connection is managed via separate interfaces, one for each IP type. LTE can support multiple PDN connections at a time, each of which can support IPv6. PDN connection setup in LTE is performed over NAS signaling, as specified by 3GPP standards.

AMSS supports LTE application profiles via PDP registry to aggregate parameters identifying a particular PDN connection. To bring up an IPv6 data call over LTE, an application must specify an LTE profile corresponding to a PDN connection that supports IPv6. Applications requesting different LTE profile numbers are bound to different interfaces and, hence, tied to different PDN connections.

8.1 IPv6 call setup

Figure 8-1 describes the high-level call flow for IPv6 call setup in LTE. In case S-GW uses GTP to initiate a PDN connection with the P-GW, the P-GW acts as the access router for the MS and allocates a globally unique /64 IPv6 prefix via a router advertisement to the MS. LTE requires stateless address autoconfiguration for IPv6 address configuration. The 64-bit IID required to form a complete IPv6 address is obtained via the PDN attach procedures. Since the network allocates and assigns the IPv6 IID to the MS, it eliminates the need for IID type configuration items. It also eliminates the need of performing any DAD, as the network ensures a unique IID allocation for the assigned prefix. Since the network assigns a globally unique IPv6 prefix to the MS, it eliminates the need for any neighbor discovery over the air interface. The following procedure corresponds to the numbered steps in Figure 8-1.

1. The MS initiates the IPv6 call bringup by initiating the PDN connection attach procedures over NAS signaling. The attach request from the MS indicates the PDN type as IPv6 or IPv4v6. The MS also includes a request for IPv6 address assignment and optionally IPv6 DNS server address assignment in the attach request.
2. On receiving the attach request from the MS, the MME requests default bearer creation by sending a create session request to the S-GW.
3. The S-GW relays the create session request to P-GW.
4. If this is the first PDN default bearer activation, the MS may need to be authenticated for access to the 3GPP network.

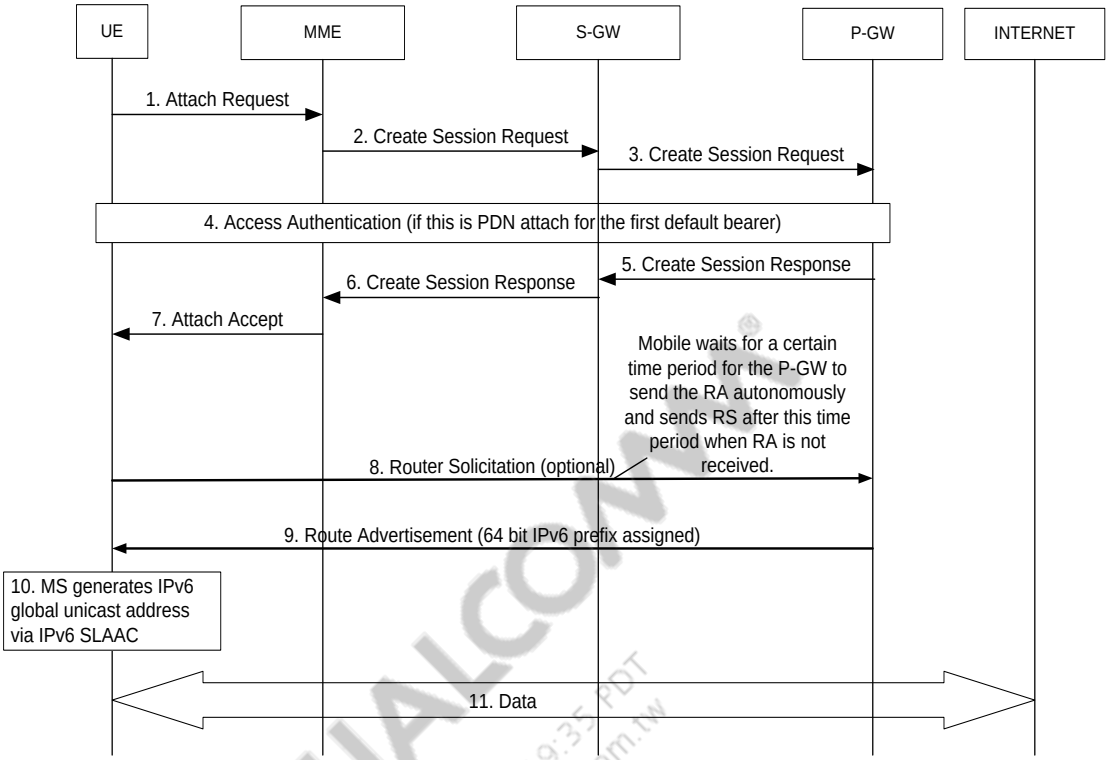

Figure 8-1 LTE IPv6 session setup

5. To allocate an IPv6 prefix to the MS, the P-GW performs radius, diameter, or DHCPv6 procedures to retrieve a globally unique IPv6 prefix for the MS. The P-GW includes this prefix along with the IPv6 IID and any allocated DNS server addresses in the create session response to the S-GW.
6. The S-GW relays the create session response from the P-GW to the MME.
7. The MME sends the allocated IPv6 IID and DNS server addresses in the attach accept message to the MS. The MME does not include the allocated IPv6 prefix in the attach accept message. If the IPv6 prefix is included, the MS ignores it.
8. Instead of sending an RS immediately after the completion of PDN attach, the MS waits for a certain time period for the P-GW to send an unsolicited RA. If the RA does not arrive within the specified timeframe, the MS sends an RS to the P-GW.
9. The P-GW sends an RA to the MS, including the unique IPv6 prefix that was allocated for the MS in earlier steps.
10. The MS generates a global unicast IPv6 address by combining the allocated IPv6 prefix and IID via the stateless address autoconfiguration procedures.
11. The MS can send/receive data using the IPv6 address configured via the above steps.

8.2 IP-agnostic applications

For LTE interfaces, if an application specifies AF_UNSPEC as the IP family type in the policy, the IP family is always mapped to the IPv4.

If the application specifies AF_UNSPEC as the IP family along with a profile number, the profile should support IPv4. Otherwise, the application request results in an error.

If the application specifies AF_UNSPEC as IP family but does not specify any profile, the default profile is chosen for the application. In such cases, if the default profile does not support IPv4 the application request results in an error.

9 Tethered Support

This chapter describes IPv6 behavior in a tethered call.

AMSS supports tethered IPv6 connectivity only via Qualcomm Modem Interface (QMI) (see [Q4]) and Remote Network (RmNet) interface (see [Q5]). IPv6 tethered connectivity is not supported via dialup networking or PPP. IPv6 connectivity over all air interface technologies is supported over QMI/RmNet.

To provide IPv6 connectivity independent of the host operating system on the tethered device, the MS acts as an IPv6 access router to the tethered device. This allows the tethered device to configure an IPv6 address via a standard compliant stateless address autoconfiguration procedure. The MS sends RAs to the tethered device to aid in stateless address autoconfiguration. The MS also processes and responds to the DAD and neighbor discovery messages from the device.

9.1 Tethered IPv6 call setup

Figure 9-1 illustrates the high-level call flow for IPv6 call setup for a device tethered to the MS. The TE is required to support QMI/RmNet for getting IPv6 connectivity. The TE can initiate IPv6 call bringup on wireless link of choice by triggering QMI signaling. The RmNet device enumerates as an IPv6 network interface on the TE provided via QMI. The host operating system on the tethered device can then configure the RmNet interface with an IPv6 address by performing stateless address autoconfiguration procedures with the MS. It is imperative for the MS to complete IPv6 call bringup over the air interface before it can act as a router for the TE. The IPv6 prefix advertised to the TE in RA is the same prefix that the MS obtains over the air link. The MS also aids with DAD and neighbor discovery procedures for the TE.

The following procedure corresponds to the numbered steps in Figure 9-1.

1. The application on TE issues a request to the MSM to start the RmNet data call using a QMI Wireless Data Service (WDS) message. Typically, a QMI-based connection manager application performs this operation.
2. The above request is handled by the QMI WDS on the MS, and it in turn initiates the procedure to bring up the data call over the air link (OTA).
3. Once the IPv6 data call is up OTA, the MS sends a response indicating success to the control point that initiated the request. In case of call origination failure, the MS sends a failure response and the following steps do not occur.

NOTE: At this point, a successful response means only that the data call originated successfully. The application still must wait for an IP address to be assigned to the TE (see the following steps) before it can begin data transfer.

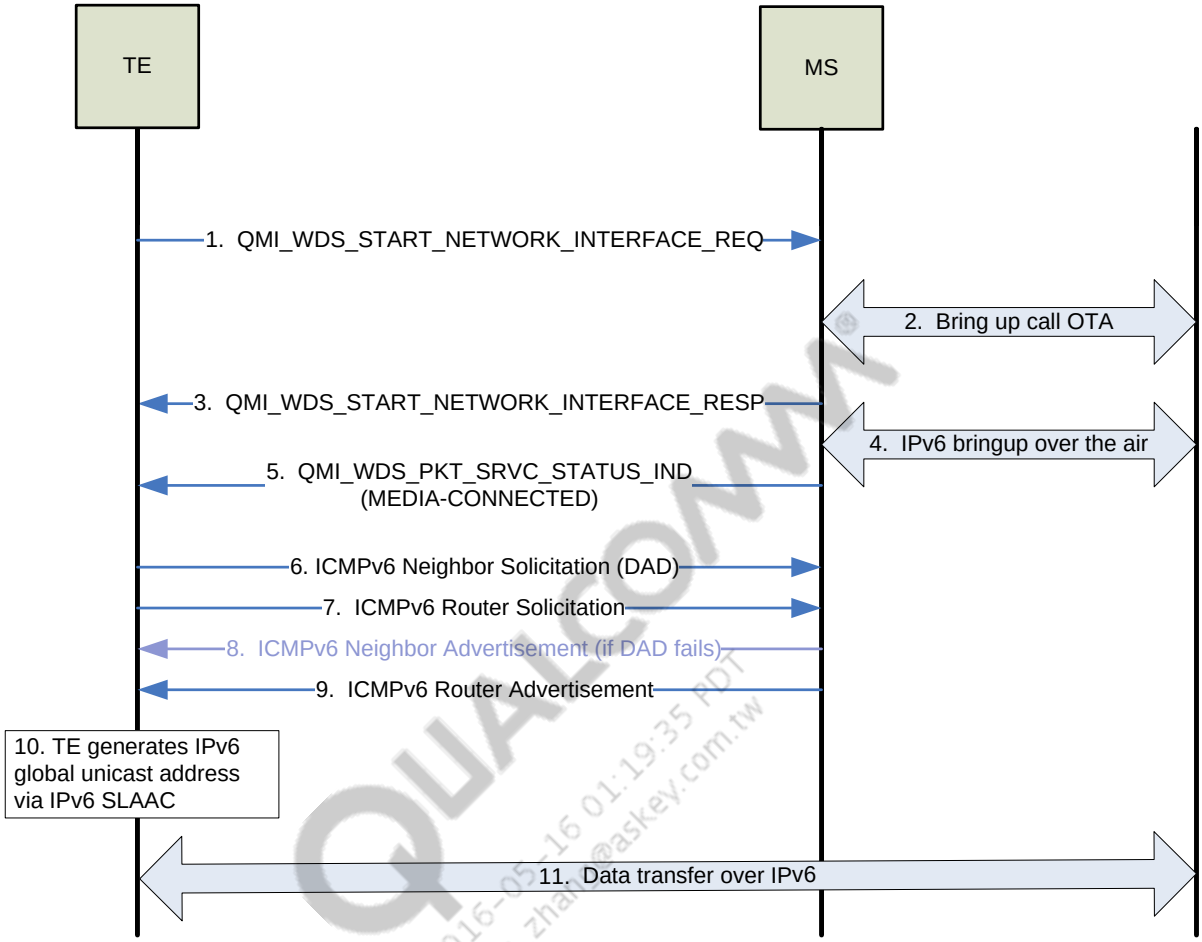

Figure 9-1 Tethered IPv6 call setup

4. To provide IPv6 connectivity to the TE, the MS completes IPv6 address configuration procedures over the wireless link. At this point, the IPv6 call is up over the air link for the MS.
5. The QMI WDS service broadcasts an indication to all its control points, indicating that the IPv6 data call is connected. The QMUX driver on the TE (which is a WDS control point) also receives this indication. The QMUX driver then notifies the OS running on the TE that the TE now has data connectivity.
6. The TE attempts to verify the uniqueness of its Interface Identifier (IID) by sending an ICMPv6 neighbor solicitation for duplicate address detection.
7. The TE requests an IPv6 prefix from the mobile by soliciting an RA.
8. If the IID requested by the TE conflicts with one already in use, the mobile responds with a neighbor advertisement indicating duplicate address detection failed. In such scenario, the TE can generate a new IID and restart the IPv6 address configuration from step 6 above.
9. The mobile responds to the RS with an RA that provides the TE with its IPv6 prefix.
10. The TE generates a global unicast IPv6 address by combining the allocated IPv6 prefix and IID via the stateless address autoconfiguration procedures.

11. The TE can now send/receive IPv6 data using the IPv6 address configured via the above steps.

NOTE: Stateless address autoconfiguration provides a standard mechanism of IPv6 address configuration. If one does not want to use the standard method, steps 6 to 11 may be skipped and instead the IPv6 address may be retrieved from the MSM using the QMI_WDS_GET_ RUNTIME_SETTINGS message and then statically configured in the TE.

9.2 IPv6 DNS server address configuration for tethered device

During the IPv6 call setup over the air interface, the MS can obtain IPv6 DNS server addresses via technology-specific signaling or via standard compliant procedures, e.g., DHCPv6. The IPv6 DNS server addresses may also be statically configured on the MS. The TE can obtain these DNS server addresses via QMI_WDS_GET_RUNTIME_SETTINGS messaging. See [Q6] for details.

AMSS does not currently support DHCPv6 server operation for DNS server address discovery for the TE. Also, AMSS currently does not support DNS address discovery via RAs.

9.3 Simultaneous tethered and embedded IPv6 calls

Simultaneous tethered and embedded IPv6 calls over the same IP interface are not supported. Simultaneous embedded and tethered IPv6 calls over separate PDN connections (LTE/eHRPD) or separate PDP contexts (UMTS) are supported, as separate PDN connections (or PDN contexts) are served over separate IP interfaces.