# IPv6 Consortium

## Core Operations Test Report Revision 1.0

**InterOperability Lab — 121 Technology Drive, Suite 2 — Durham, NH 03824 -(603) 862-2804**

| | | |
|---|---|---|
| Technical Manager: | Erica Williamsen | ericaw@iol.unh.edu |
| Technician: | John Tester | jtester@iol.unh.edu |


March 18, 2005

Joe Contact
Company A
1 Main Street
Durham, NH 03820

Mr. Joe Contact,

Enclosed are the results from the IPv6Ready PhaseII Logo Base Specification testing performed on:

> Device A. MAC Address aa-bb-cc-dd-ee-ff. Console "show config" command reports software version Release abcd.

This testing pertains to a set of standard requirements, put forth in RFC2460-2463, 1981 and 2373. The tests performed are part of the IPv6Ready PhaseII Base Specification Test Suite, which is available on the UNH InterOperability Lab's website:

ftp://ftp.iol.unh.edu/pub/ipv6/testsuites/IPv6_Ready_Test_Specification_Core_Protocols.pdf

As always, we welcome any comments regarding this Test Suite.

During the testing process, the following issues were uncovered:

| Test | Result |
|---|---|
| v6LC.1.2.15.A | The RUT sent an ICMPv6 Parameter Problem message with an incorrect Pointer field of 0x1 after receiving a packet with a Routing Header with an odd Header Extension Length. |
| v6LC.1.2.15.E | The RUT sent a Destination Unreachable message instead of discarding the echo request from TN2 and transmitting an ICMP Time Exceeded – Hop Limit Exceeded message. |
| v6LC.1.2.15.F | The RUT sent a Destination Unreachable message instead of discarding the echo request from TN2 and transmitting an ICMP Time Exceeded – Hop Limit Exceeded message. |
| v6LC.2.1.6.A,B | The Retransmit Timer variable could not be configured by system management. |
| v6LC.2.1.18.I,K,O | After receiving a Neighbor Advertisement while in state REACHABLE with certain bits set, the NUT did not change to the new state STALE and did not transmit a unicast Neighbor Solicitation after DELAY_FIRST_PROBE_TIME. |
| v6LC.2.2.8.B | The RUT transmitted Router Advertisements with an incorrect Router Lifetime field of 120 seconds. |
| v6LC.2.2.9.B | The RUT transmitted Router Advertisements more than once every 3 seconds. |
| v6LC.3.1.3.D,H | The NUT did not send a solicited Neighbor Advertisement for it's autoconfigured link-local address, indicating that the NUT did not ignore the received DAD Neighbor Solicitation. |
| v6LC.4.1.6.A,B | The NUT did not include a Fragment header in the Echo Reply after receiving a Packet Too Big message from TR1 |



If you have any questions about the test procedures or results, please feel free to contact me via e-mail at jtester@iol.unh.edu or by phone at 603-862-2804.

Regards,

John Tester

The following table contains the test results and their meanings.

| Result | Interpretation |
|---|---|
| **PASS** | The NUT was observed to exhibit conformant behavior. |
| **FAIL** | The NUT was observed to exhibit non-compliant behavior. |
| **PASS with Comments** | The NUT was observed to exhibit conformant behavior, however there are additional comments supplied. |
| **Warning** | The NUT was observed to exhibit behavior that is not recommended. |
| **Refer to Comments** | From the observations, a valid pass or fail could not be determined. An additional explanation of the situation is included. |
| **Not Applicable (N/A)** | The NUT does not support the technology required to perform these tests. |
| **Not Available** | Due to testing station or time limitations, the tests could not be performed, or were performed in a limited capacity. |
| **Not Tested (N/T)** | Not tested due to time constraint of the test period. |
| **Borderline** | The observed values of the parameter is valid at one extreme, and invalid at the other extreme. |
| **Informative** | Results are for informative purposes only and are not judged on a pass or fail basis. |

# Section 1: RFC 2460

These tests are designed to verify the readiness of an IPv6 implementation vis-à-vis the IPv6 Base specification.

## Group 1: IPv6 Header

The following tests cover the fields in the basic IPv6 header.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.1.1 | **Version Field** | **A** | **PASS** |
| **Purpose**: Verify that a node properly processes the Version field of received packets. | | | |
| **Comments on Test Procedure** | | | |
| A. TN1 transmits an ICMPv6 Echo Request to the NUT with version fields of 4, 0, 5, 7, and 15, with each packet followed by a valid ICMPv6 Echo Request. | | | |
| **Comments on Test Results** | | | |
| A. The NUT did not crash or generate invalid packets. The NUT responded to the second Echo Request from TN1, with a valid version field of 6. | | | |

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.1.2 | **Traffic Class Non-Zero –End Node** | **A** | **PASS** |
| **Purpose**: Verify that a node properly processes the Traffic Class field of received packets and generates a valid value in transmitted packets. | | | |
| **Comments on Test Procedure** | | | |
| A. TN1 transmits an ICMPv6 Echo Request with a Traffic Class field of 32, a non-zero value to the NUT. | | | |
| **Comments on Test Results** | | | |
| A. The NUT generated an Echo Reply. The Traffic Class field is zero. The NUT does not support a specific use of the Traffic Class field, so the Traffic Class field in the Echo Reply is zero. | | | |

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.1.3 | **Traffic Class Non-Zero –Intermediate Node (Routers Only)** | **A** | **PASS** |

**Purpose**: Verify that a router properly processes the Traffic Class field of received packets and generates a valid value in transmitted packets.

**Comments on Test Procedure**

A. TN1 transmits an ICMPv6 Echo Request with a Traffic Class field of 32, a non-zero value to TN2, with a first hop through the RUT.

**Comments on Test Results**

A. The RUT forwarded the Echo Request. The Traffic Class field is passed on to TN2 unchanged. The RUT does not support a specific use of the Traffic Class field, so the Traffic Class field in the Echo Request is unchanged.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.1.4 | **Flow Label Non-Zero** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a node properly processes the Flow Label field of received packets and generates a valid value in transmitted packets.

**Comments on Test Procedure**

A. *NUT receives Non-Zero Flow Label:* TN1 transmits an ICMPv6 Echo Request with a Flow Label of 0x34567 to the NUT.
B. *RUT forwards Non-Zero Flow Label (Routers Only):* TN1 transmits an ICMPv6 Echo Request with a Flow Label 0x34567 to TN2's Global Address with a first hop through the RUT.

**Comments on Test Results**

A. The NUT generated an Echo Reply. The Flow Label field is zero. The NUT does not support use of the Flow Label field, the Flow Label in the Echo Reply is zero.
B. The RUT forwarded the Echo Reply from TN1 to TN2. The RUT does not support the use of the Flow Label field; the field is unchanged in the forwarded packet.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.1.5 | **Payload Length** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a node properly processes the Payload Length field of received packets.

**Comments on Test Procedure**

A. *Payload Length Odd*: TN1 transmits an ICMPv6 Echo Request that has an IPv6 header with a Payload Length of 0x33 (51) to the NUT.
B. *Payload Length Even:* TN1 transmits an ICMPv6 Echo Request that has an IPv6 header with a Payload Length of 0x32 (50) to the NUT.

**Comments on Test Results**

A. The NUT generated an Echo Reply, indicating successful processing of the packet.
B. The NUT generated an Echo Reply, indicating successful processing of the packet.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.1.6 | **No Next Header after IPv6 Header** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify proper behavior of a node when it encounters a Next Header value of 59 (no next header).

**Comments on Test Procedure**

A. *NUT Receives No Next Header:* TN1 transmits an ICMPv6 Echo Request to the NUT, which contains an IPv6 header with a Next Header of 59 and an ICMPv6 Echo Request Header following the IPv6 header.
B. *RUT Forwards No Next Header (Routers Only):* TN1 transmits an ICMPv6 Echo Request containing an IPv6 header with a Next Header of 59 to TN2's Global Address with a first hop through the RUT.

**Comments on Test Results**

A. The NUT did not send any packets in response to Packet A.
B. The RUT forwarded the Echo Request to TN2. The octets after the IPv6 header with a Next Header field of 59 (the ICMPv6 Request octets) were unchanged.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.1.7 | **Unrecognized Next Header** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a node generates the appropriate response to an unrecognized or unexpected Next Header field.

**Comments on Test Procedure**

A. *Unrecognized Next Header in IPv6 Header (Multiple Values):* TN1 transmits an ICMPv6 Echo Request to the NUT, which has an IPv6 header with a Next Header field of 136. This is followed by a valid Echo Request, destined for the NUT. Repeat with all unrecognized Next Header values between 137 and 254.
B. *Unexpected Next Header in IPv6 Header:* TN1 transmits an ICMPv6 Echo Request to the NUT, which has an IPv6 header with a Next Header field of 0. The actual extension header that follows is a Fragment header. The Fragment ID is 135.

**Comments on Test Results**

A. The NUT sent an ICMPv6 Parameter Problem message to TN1 with a code field of 1 (unrecognized Next Header type encountered), and a pointer field of 0x06 (offset of the Next Header field). The NUT responded to the valid Echo Requests that follow each version of the Echo Requests with unrecognized next header values.
B. The NUT sent an ICMPv6 Parameter Problem message to TN1 with a code field of 2 (unrecognized IPv6 Option encountered) and a Pointer field of 0x2e (offset of the Fragment ID in the Fragment Header). The NUT discarded the Echo Request and did not send a reply to TN1.



| Test # | | Result | |
|---|---|---|---|
| v6LC.1.1.8 | **Hop Limit Zero – End Node** | **A** | **PASS** |

**Purpose**: Verify that a node correctly processes the Hop Limit field of received packets and generates a valid value in transmitted packets.

**Comments on Test Procedure**

A. TN1 transmits an ICMPv6 Echo Request with a Hop Limit field of zero to the NUT.

**Comments on Test Results**

A. The NUT generated an Echo Reply with a Hop Limit field value of greater than zero.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.1.9 | **Hop Limit Decrement – Intermediate Node (Routers Only)** | **A** | **PASS** |

**Purpose**: Verify that a router correctly processes the Hop Limit field of received packets and generates a valid value in transmitted packets.

**Comments on Test Procedure**

A. TN1 transmits an ICMPv6 Echo Request to TN2's Global Address with a first hop through the RUT. The Hop Limit field is set to 15.

**Comments on Test Results**

A. The RUT forwarded the Echo Request to TN2 with the Hop Limit field decremented to 14.

## Group 2: Extension Headers and Options

The following tests cover the processing of options and extension headers, particularly the Hop-by-Hop Options, Destination Options, and Routing headers.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.1 | **Next Header Zero** | **A** | **PASS** |
| **Purpose**: Verify that a node discards a packet that has a Next Header field of zero in a header other than an IPv6 header and generates an ICMPv6 Parameter Problem message to the source of the packet. | | | |
| **Comments on Test Procedure** | | | |
| A. TN1 transmits an ICMPv6 Echo Request to the NUT, which has a Hop-by-Hop Options header with a Next Header field of zero. | | | |
| **Comments on Test Results** | | | |
| A. The NUT sent an ICMPv6 Parameter Problem message to TN1 with a Code field of 1 (unrecognized Next Header type encounter), and a Pointer field of 0x28 (offset of the Next Header field of the Hop-by Hop Options header). The NUT discarded the Echo Request and did not send an Echo Reply to TN1. | | | |

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.2 | **No Next Header after Extension Header** | **A** | **PASS** |
| | | **B** | **PASS** |
| **Purpose**: Verify proper behavior of a node when it encounters a Next Header value of 59 (no next header). | | | |
| **Comments on Test Procedure** | | | |
| A. *End Node*: TN1 transmits an ICMPv6 Echo Request to the NUT, which contains a Destination Options header with a Next Header of 59.<br>B. *Intermediate Node (Routers Only):* TN1 transmits an ICMPv6 Echo Request to TN2, which contains a Destination Options header with a Next Header of 59, with a first hop through the RUT. | | | |
| **Comments on Test Results** | | | |
| A. The NUT did not send any packets in response to the Echo Request.<br>B. The RUT forwarded the Echo Request to TN2 on Link A. The octets past the end of the header whose Next Header field contains 59 are unchanged. | | | |

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.3 | **Unrecognized Next Header in Extension Header – End Node** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a node discards a packet with an unrecognized or unexpected next header in an extension header and transmits an ICMPv6 Parameter Problem message to the source of the packet.

**Comments on Test Procedure**

A. *Unrecognized Next Header in Extension Header (Multiple Values):* TN1 transmits an IPv6 packet, which has a Destination Options header with a Next Header field of 136, followed by a valid Echo Request to the NUT. This is repeated with all unrecognized Next Header values between 137 and 254.
B. *Unexpected Next Header in Extension Header:* TN1 transmits an ICMPv6 Echo Request, which has a Destination Options header with a Next Header field of 60. The actual extension header that follows is a Fragment header, whose offset is 0x10E0 (so that the first 8 bits of this 13 bit field would be 135). The second reserved field is 0x2 and the more bit is clear. (If processed as a Destination Options header, this would be processed as Option Data Length equals 4.)

**Comments on Test Results**

A. The NUT sent an ICMPv6 Parameter Problem message to TN1. The ICMPv6 Code field was 1 (unrecognized Next Header type encountered). The ICMPv6 Pointer field was 0x28 (offset of the Next Header field). The NUT sent an Echo Reply in response to the valid Echo Requests sent by TN1 after each unrecognized Next Header packet (Packet A).
B. From the Next Header field in the Destination Options header, the NUT expects the Fragment header to be a Destination Options header. Thus, the Fragment Offset would be interpreted as if it were an Option Type. The NUT sent an ICMPv6 Parameter Problem Message to TN1. The Code field was 2 (unrecognized IPv6 Option encountered). The Pointer field was 0x32 (offset of the Fragment Offset in the Fragment header). The NUT discarded the Echo Request and did not send an Echo Reply to TN1.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.4 | **Extension Header Processing Order** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |

**Purpose**: Verify that a node properly processes the headers of an IPv6 packet in the correct order.

**Comments on Test Procedure**

A. *Destination Options Header precedes Fragment Header, Error from Destination Options Header:* TN1 transmits an Echo Request that has a Hop-by-Hop Options header, Destination Options header, and Fragment header, in that order. The Destination Options header has an unknown Option Type of 135. The IPv6 header has a Payload Length that is not a multiple of 8 octets, and the Fragment header has the M-bit set.

B. *Destination Options Header precedes Fragment Header, Error from Fragment Header:* TN1 transmits an Echo Request that has a Hop-by-Hop Options header, Destination Options header, and Fragment header, in that order. The Destination Options header has an unknown Option Type of 7. The IPv6 header has a Payload Length that is not a multiple of 8 octets, and the Fragment header has the M-bit set.

C. *Fragment Header precedes Destination Options Header, Error from Fragment Header:* TN1 transmits an Echo Request that has a Hop-by-Hop Options header, Fragment header, and Destination Options header, in that order. The IPv6 header has a Payload Length that is not a multiple of 8 octets, and the Fragment header has the M-bit set. The Destination Options header has an unknown Option Type of 135.

D. *Fragment Header precedes Destination Options Header, Error from Destination Options Header:* TN1 transmits an Echo Request that has a Hop-by-Hop Options header, Fragment header, and Destination Options header, in that order. The IPv6 header has a Payload Length that is not a multiple of 8 octets, and the Fragment header does not have the M-bit set. The Destination Options header has an unknown Option Type of 135.

**Comments on Test Results**

A. The NUT sent an ICMPv6 Parameter Problem message to TN1 with a Code field of 2 (Unrecognized IPv6 Option Encountered), and Pointer field of 0x32 (offset of the Option type field in the Destination Options Header). The NUT discarded the Echo Request from TN1.

B. The NUT sent an ICMPv6 Parameter Problem message to TN1 with a Code field of 0 (erroneous header field encountered), and Pointer field of 0x04 (offset of the Payload Length field in the IPv6 Header). The NUT discarded the Echo Request from TN1.

C. The NUT sent an ICMPv6 Parameter Problem message to TN1 with a Code field of 0 (erroneous header field encountered), and Pointer field of 0x04 (offset of the Payload Length field in the IPv6 Header). The NUT discarded the Echo Request from TN1.

D. The NUT sent an ICMPv6 Parameter Problem message to TN1 with a Code field of 2 (Unrecognized IPv6 Option Encountered), and Pointer field of 0x3A (offset of the Option type field in the Destination Options Header). The NUT discarded the Echo Request from TN1.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.5 | **Option Processing Order** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |

**Purpose**: Verify that a node properly processes the options in a single header in the order of occurrence.

**Comments on Test Procedure**

A. *First Option has Most Significant Bits $00_b$, Next has Most Significant Bits $01_b$*: TN1 transmits an Echo Request to the NUT that has a Destination Options header with four unknown Options. The Option Types are 7, 71, 135, and 199.
B. *First Option has Most Significant Bits $00_b$, Next has Most Significant Bits $10_b$*: TN1 transmits an Echo Request to the NUT that has a Destination Options header with four unknown Options. The Option Types are 7, 135, 199, and 71.
C. *First Option has Most Significant Bits $00_b$, Next has Most Significant Bits $11_b$*: TN1 transmits an Echo Request to the NUT's Link-Local address that has a Destination Options header with four unknown Options. The Option Types are 7, 199, 71, and 135.

**Comments on Test Results**

A. The NUT silently discarded the ICMPv6 Echo Request and did not send any Packets to TN1.
B. The NUT sent an ICMPv6 Parameter Problem message to TN1 with a Code field of 2 (unrecognized IPv6 Option encountered) and a Pointer field of 0x30 (offset of the Option Type field of the second option). The NUT discarded the Echo Request sent by TN1 and did not send a Reply.
C. The NUT sent an ICMPv6 Parameter Problem message to TN1 with a Code field of 2 (unrecognized IPv6 Option encountered) and a Pointer field of 0x30 (offset of the Option Type field of the second option). The NUT discarded the Echo Request sent by TN1 and did not send a Reply.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.6 | **Option Processing, Hop-by-Hop Options Header – End Node** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |
| | | **G** | **PASS** |
| | | **H** | **PASS** |

**Purpose**: Verify that a node properly processes both known and unknown options, and acts in accordance with the highest order two bits of the option.

**Comments on Test Procedure**

A. *Pad1 Option:* TN1 transmits an ICMPv6 Echo Request that has a Hop-by-Hop Options header with six Pad1 Options
B. *PadN Option:* TN1 transmits an ICMPv6 Echo Request that has a Hop-by-Hop Options header with a 6 byte PadN Option.
C. *Most Significant Bits $00_b$:* TN1 transmits an ICMPv6 Echo Request that has a Hop-by-Hop Options header with an unknown Option Type of 7.
D. *Most Significant Bits $01_b$:* TN1 transmits an ICMPv6 Echo Request that has a Hop-by-Hop Options header with an unknown Option Type of 71.
E. *Most Significant Bits $10_b$, unicast destination:* TN1 transmits an ICMPv6 Echo Request that has a Hop-by-Hop Options header with an unknown Option Type of 135.
F. *Most Significant Bits $11_b$, unicast destination:* TN1 transmits an ICMPv6 Echo Request that has a Hop-by-Hop Options header with an unknown Option Type of 199.
G. *Most Significant Bits $10_b$, multicast destination:* TN1 transmits an ICMPv6 Echo Request to a local multicast address that has a Hop-by-Hop Options header with an unknown Option Type of 135.
H. *Most Significant Bits $11_b$, multicast destination:* TN1 transmits an ICMPv6 Echo Request to a local multicast address that has a Hop-by-Hop Options header with an unknown Option Type of 199.



**Comments on Test Results**

A. The NUT sent an Echo Reply to TN1.
B. The NUT sent an Echo Reply to TN1.
C. The NUT skipped the unknown option and the header was processed. The NUT sent an Echo Reply to TN1.
D. The NUT did not generate any packets sent to TN1. The NUT discarded the Echo Request.
E. The NUT sent an ICMPv6 Parameter Problem message to TN1. The Code field was 2 (unrecognized IPv6 Option encountered) and the Pointer field was 0x2A (offset of the option field of Hop-by -Hop Options header). The NUT discarded the Echo Request, and did not send and Echo Reply. The invoking Echo Request packet contained in the Parameter Problem message did not exceed the minimum Ipv6 MTU, and:
    - The Source Address of the Parameter Problem Message was the same as the Destination Address in TN1's Echo Request Packet.
    - The Destination Address was the same as the Source Address in TN1's Echo Request Packet.

F. The NUT sent an ICMPv6 Parameter Problem message to TN1. The Code field was 2 (unrecognized IPv6 Option encountered) and the Pointer field was 0x2A (offset of the option field of Hop-by -Hop Options header). The NUT discarded the Echo Request, and did not send and Echo Reply. The invoking Echo Request packet contained in the Parameter Problem message did not exceed the minimum Ipv6 MTU, and:
    - The Source Address of the Parameter Problem Message was the same as the Destination Address in TN1's Echo Request Packet.
    - The Destination Address was the same as the Source Address in TN1's Echo Request Packet.

G. The NUT sent an ICMPv6 Parameter Problem message to TN1. The Code field was 2 (unrecognized IPv6 Option encountered) and the Pointer field was 0x2A (offset of the option field of Hop-by -Hop Options header). The NUT discarded the Echo Request, and did not send and Echo Reply. The invoking Echo Request packet contained in the Parameter Problem message did not exceed the minimum Ipv6 MTU, and:
   - The Destination Address of the Parameter Problem Message was the same as the Source Address in TN1's Echo Request Packet.

H. The NUT did not generate any packets sent to TN1. The Echo Request was discarded, as the destination address was multicast. The NUT did not send an ICMPv6 Parameter Problem message.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.7 | **Options Processing, Hop-by-Hop Options Header – Intermediate Node (Routers Only)** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |
| | | **G** | **PASS** |
| | | **H** | **PASS** |

**Purpose**: Verify that a router properly processes both known and unknown options, and acts in accordance with the highest order two bits of the option.

**Comments on Test Procedure**

A. *Pad1 Option:* TN1 transmits an ICMPv6 Echo Request that has a Hop-by-Hop Options header with six Pad1 Options
B. *PadN Option:* TN1 transmits an ICMPv6 Echo Request that has a Hop-by-Hop Options header with a 6 byte PadN Option.
C. *Most Significant Bits $00_b$:* TN1 transmits an ICMPv6 Echo Request that has a Hop-by-Hop Options header with an unknown Option Type of 7.
D. *Most Significant Bits $01_b$:* TN1 transmits an ICMPv6 Echo Request that has a Hop-by-Hop Options header with an unknown Option Type of 71.
E. *Most Significant Bits $10_b$, unicast destination:* TN1 transmits an ICMPv6 Echo Request that has a Hop-by-Hop Options header with an unknown Option Type of 135.
F. *Most Significant Bits $11_b$, unicast destination:* TN1 transmits an ICMPv6 Echo Request that has a Hop-by-Hop Options header with an unknown Option Type of 199.
G. *Most Significant Bits $10_b$, multicast destination:* TN1 transmits an ICMPv6 Echo Request to a local multicast address that has a Hop-by-Hop Options header with an unknown Option Type of 135.
H. *Most Significant Bits $11_b$, multicast destination:* TN1 transmits an ICMPv6 Echo Request to a local multicast address that has a Hop-by-Hop Options header with an unknown Option Type of 199.



**Comments on Test Results**

A. The RUT must forward the Echo Request to TN2.
B. The RUT must forward the Echo Request to TN2.
C. The unknown option is skipped and the header is processed. The RUT must forward the Echo Request to TN2.
D. The RUT must not forward the Echo Request to TN2. The Echo Request is discarded.
E. The RUT sent an ICMPv6 Parameter Problem message to TN1. The Code field was 2 (unrecognized IPv6 Option encountered) and the Pointer field was 0x2A (offset of the option field of Hop-by -Hop Options header). The RUT discarded the Echo Request, and did not forward it to TN2. The invoking Echo Request packet contained in the Parameter Problem message did not exceed the minimum Ipv6 MTU, and:
   - The Destination Address was the same as the Source Address in TN1's Echo Request Packet.
F. The RUT sent an ICMPv6 Parameter Problem message to TN1. The Code field was 2 (unrecognized IPv6 Option encountered) and the Pointer field was 0x2A (offset of the option field of Hop-by -Hop Options header). The RUT discarded the Echo Request, and did not forward it to TN2. The invoking Echo Request packet contained in the Parameter Problem message did not exceed the minimum Ipv6 MTU, and:
   - The Destination Address was the same as the Source Address in TN1's Echo Request Packet.
G. The RUT sent an ICMPv6 Parameter Problem message to TN1. The Code field was 2 (unrecognized IPv6 Option encountered) and the Pointer field was 0x2A (offset of the option field of Hop-by -Hop Options header). The RUT discarded the Echo Request, and did not forward it to TR1. The invoking Echo Request packet contained in the Parameter Problem message did not exceed the minimum Ipv6 MTU.
H. The RUT must not forward the Echo Request to TR1. The Echo Request is discarded, as the destination address is multicast. The RUT must not send an ICMPv6 Parameter Problem message.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.8 | **Option Processing, Destination Options Header** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |
| | | **G** | **PASS** |
| | | **H** | **PASS** |

**Purpose**: Verify that a node properly processes both known and unknown options, and acts in accordance with the highest order two bits of the option.

**Comments on Test Procedure**

A. *Pad1 Option*: TN1 transmits an Echo Request, that has a Destination Options header with six Pad1 Options, to the NUT.
B. *PadN Option*: TN1 transmits an Echo Request, that has a Destination Options header with a 4 byte PadN Option, to the NUT.
C. *Most Significant Bits $00_b$*: TN1 transmits an Echo Request, that has a Destination Options header with an unknown Option type of 7, to the NUT.
D. *Most Significant Bits $01_b$*: TN1 transmits an Echo Request, that has a Destination Options header with an unknown Option type of 71, to the NUT.
E. *Most Significant Bits $10_b$, unicast destination*: TN1 transmits an Echo Request, that has a Destination Options header with an unknown Option type of 135, to the NUT.
F. *Most Significant Bits $11_b$, unicast destination:* TN1 transmits an Echo Request, that has a Destination Options header with an unknown Option type of 199, to the NUT.
G. *Most Significant Bits $10_b$, multicast destination*: TN1 transmits an Echo Request, sent to a local multicast address that has a Destination Options header with an unknown Option Type of 135, to the NUT.
H. *Most Significant Bits $11_b$, multicast destination*: TN1 transmits an Echo Request, sent to a local multicast address that has a Destination Options header with an unknown Option Type of 199, to the NUT.



**Comments on Test Results**

A. The NUT sent an Echo Reply to TN1.
B. The NUT sent an Echo Reply to TN1.
C. The unknown option in the Destination Options header was skipped and the header was processed. The NUT sent an Echo Reply to TN1.
D. The NUT did not generate any packets sent to TN1. The Echo Request was discarded and not replied to.
E. The NUT sent and ICMPv6 Parameter Problem message to TN1 with a Code field of 2 (unrecognized IPv6 Option encountered) and a Pointer field of 0x2A (offset of the option field of Destination Options header). The NUT discarded the Echo Request and did not send a Reply. The invoking Echo Request packet included in the Error Message did not exceed the minimum IPv6 MTU, in addition:
    - The Source Address of the Parameter Problem Message was the same as the Destination Address in the Echo Request Packets sent from TN1.
    - The Destination Address was the same as the Source Address in the Echo Request Packet sent from TN1.
F. The NUT sent and ICMPv6 Parameter Problem message to TN1 with a Code field of 2 (unrecognized IPv6 Option encountered) and a Pointer field of 0x2A (offset of the option field of Destination Options header). The NUT discarded the Echo Request and did not send a Reply. The invoking Echo Request packet included in the Error Message did not exceed the minimum IPv6 MTU, in addition:
    - The Source Address of the Parameter Problem Message was the same as the Destination Address in the Echo Request Packets sent from TN1.

- The Destination Address was the same as the Source Address in the Echo Request Packet sent from TN1.

G. The NUT sent and ICMPv6 Parameter Problem message to TN1 with a Code field of 2 (unrecognized IPv6 Option encountered) and a Pointer field of 0x2A (offset of the option field of Destination Options header). The NUT discarded the Echo Request and did not send a Reply. The invoking Echo Request packet included in the Error Message did not exceed the minimum IPv6 MTU, in addition:

- The Destination Address in the Parameter Problem message was the same as the Source Address in the Echo Request Packet sent from TN1.

H. The NUT did not generate any packets sent to TN1. The Echo Request was discarded and not replied to, since the destination address was multicast. The NUT did not send an ICMPv6 Parameter Problem message.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.9 | **Responding to Routing Header - End Node** | **A** | **PASS** |

**Purpose**: Verify that a node properly responds to an IPv6 packet destined for it that contains a Routing header.

**Comments on Test Procedure**

A. If the NUT is a Host, TR1 transmits a Router Advertisement to the all-nodes multicast address (FF02::1). The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. The Router and Prefix Lifetimes are long enough such that they do not expire during the test. If the NUT is a router, configure a default route with TR1 as the next hop. TR1 then transmits an Echo Request to the NUT. Observe the packets sent by the NUT. TR1 transmits a Neighbor Advertisement. After an Echo Reply is received, TR1 forwards an Echo Request that has a Routing Header with three segments which is destined for the NUT.

**Comments on Test Results**

A. The NUT added TR1 to its Default Router List, configured a global address, and computed Reachable Time. The NUT Solicited TR1 and sent an Echo Reply to TR1, no other packets were sent, indicating the NUT's NCE for TR1 is in state **REACHABLE**. The NUT transmitted an Echo Reply to TN2's Global Address using TR1 as a first hop. The Echo Reply did not contain a Routing Header.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.10 | **Unrecognized Routing Type - End Node** | **A** | **PASS** |

**Purpose**: Verify that a node properly processes an IPv6 packet destined for it that contains a Routing header with an unrecognized Routing Type value.

**Comments on Test Procedure**

A. If the NUT is a Host, TR1 transmits a Router Advertisement to the all-nodes multicast address (FF02::1). The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. The Router and Prefix Lifetimes are long enough such that they do not expire during the test. If the NUT is a router, configure a default route with TR1 as the next hop. TR1 then transmits an Echo Request to the NUT. Observe the packets sent by the NUT. TR1 transmits a Neighbor Advertisement. After an Echo Reply is received, TR1 forwards an Echo Request that has a Routing Header with a Routing Type value of 33 and a Segments Left value of 0. The Echo Request is destined for the NUT.

**Comments on Test Results**

A. The NUT added TR1 to its Default Router List, configured a global address, and computed Reachable Time. The NUT Solicited TR1 and sent an Echo Reply to TR1, no other packets were sent, indicating the NUT's NCE for TR1 is in state **REACHABLE**. The NUT ignored the unrecognized Routing Type value and responded to the Echo Request with an Echo Reply sent to TN2 with TR1 as the first-hop.



| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.11 | **Unrecognized Routing Type - Intermediate Node (Routers Only)** | **A** | **PASS** |

**Purpose**: Verify that a router properly processes an IPv6 packet as the intermediate node that contains a Routing header with an unrecognized Routing Type value.

**Comments on Test Procedure**

A. Configure a default route with TR1 as the next hop. TR1 and TR2 transmit an Echo Request to the RUT. Observe the packets sent by the RUT. TR1 and TR2 transmit a Neighbor Advertisement. After an Echo Reply is received, TN2 transmits an Echo Request that has a Routing Header with a Routing Type value of 33 and Segments Left value of 1. The Echo Request is destined for TN1 and has a first hop through the RUT.

**Comments on Test Results**

A. The RUT should have discarded the Echo Request from TN2 and transmitted an ICMP Parameter Problem message, Code 0, and a pointer field of 0x2A (the offset of the Routing type field of the Routing Header) to the global address of TN2.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.12 | **Routing Header Reserved Field – End Node** | **A** | **PASS** |

**Purpose**: Verify that a node properly processes an IPv6 packet destined for it that contains a Routing header with a non-zero Reserved field.

**Comments on Test Procedure**

A. If the NUT is a Host, TR1 transmits a Router Advertisement to the all-nodes multicast address (FF02::1). The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. The Router and Prefix Lifetimes are long enough such that they do not expire during the test. If the NUT is a router, configure an address with the advertised prefix. TR1 then transmits an Echo Request to the NUT. Observe the packets sent by the NUT. TR1 transmits a Neighbor Advertisement. After an Echo Reply is received, TR1 forwards, to the NUT, an Echo Request that has a Routing Header with a non-zero reserved field.

**Comments on Test Results**

A. The NUT added TR1 to its Default Router List, configured a global address, and computed Reachable Time. The NUT Solicited TR1 and sent an Echo Reply to TR1, no other packets were sent, indicating the NUT's NCE for TR1 is in state **REACHABLE**. The NUT ignored the non-zero reserved field and transmitted an Echo Reply to TN2's Global Address using TR1 as the first hop.



| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.13 | **Routing Header Reserved Field – Intermediate Node (Routers Only)** | **A** | **PASS** |

**Purpose**: Verify that a router properly processes an IPv6 packet destined for it that contains a Routing header with a non-zero Reserved field.

**Comments on Test Procedure**

A. Configure a default route with TR1 as the next hop. TR1 then transmits an Echo Request to the RUT. Observe the packets sent by the RUT. TR1 transmits a Neighbor Advertisement. After an Echo Reply is received, TN1 transmits an Echo Request to the RUT that has a Routing header with a non-zero reserved field.

**Comments on Test Results**

A. The RUT Solicited TR1 and sent an Echo Reply to TR1, no other packets were sent, indicating the RUT's NCE for TR1 is in state **REACHABLE**. The RUT ignored the non-zero reserved field and forwarded the Echo Request to TR1's Global Address. The RUT did not change the value of the reserved field when the packet was forwarded.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.14 | **Routing Header Processing – End Node** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a node properly processes an IPv6 packet destined for it that contains a Routing header.

**Comments on Test Procedure**

*Setup for each part:* If the NUT is a Host, TR1 transmits a Router Advertisement to the all-nodes multicast address (FF02::1). The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. The Router and Prefix Lifetimes are long enough such that they do not expire during the test. If the NUT is a router, configure a default route with TR1 as the next hop. TR1 then transmits an Echo Request to the NUT. Observe the packets sent by the NUT. TR1 transmits a Neighbor Advertisement. After an Echo Reply is received setup is complete.

A. *Header Extension Length Odd*: TR1 transmits an Echo Request to the NUT that has a Routing Header with an odd Header Extension Length value of 5.
B. *Address[i] is Multicast*: TN2 transmits an Echo Request to the NUT that has a Routing Header with the first address in the Routing Header as TR1's Solicited-Node Multicast Address.

**Comments on Test Results**

*Observations of Setup for Each Part:* The NUT added TR1 to its Default Router List, configured a global address, and computed Reachable Time. The NUT Solicited TR1 and sent an Echo Reply to TR1, no other packets were sent, indicating the NUT's NCE for TR1 is in state **REACHABLE**.

A. The NUT ignored the invalid field of the Routing Header and processed the Echo Request from TN1. The NUT responded to TN1 with an Echo Reply.
B. The NUT ignored the invalid field of the Routing Header and processed the Echo Request from TN1. The NUT responded to TN1 with an Echo Reply.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.2.15 | **Routing Header Processing – Intermediate Node (Routers Only)** | **A** | **FAIL** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **FAIL** |
| | | **F** | **FAIL** |
| | | **G** | **PASS** |

**Purpose**: Verify that a router properly processes an IPv6 packet destined for it that contains a Routing header.

**Comments on Test Procedure**

*Setup for each part:* Configure a default route with TR1 as the next hop. TR1 then transmits an Echo Request to the RUT. Observe the packets sent by the RUT. TR1 transmits a Neighbor Advertisement. After an Echo Reply is received setup is complete.

A. *Header Extension Length Odd*: TN2 transmits an Echo Request that has a Routing Header with an odd Header Extension Length value of 5. The Echo Request is destined for TN1 with the RUT as the first hop.
B. *Segments Left Greater Than Number of Addresses*: TN2 transmits an Echo Request that has a Routing Header with a Segments Left field that is greater than the number of addresses in the header. The Echo Request is destined for TN1 with the RUT as the first hop.
C. *Address[i] is Multicast*: TN2 transmits an Echo Request that has a Routing Header with the first address in the Routing Header as TR1's Solicited-Node Multicast Address. The Echo Request is destined for TN1 with the RUT as the first hop.
D. *Destination is Multicast*: TN2 transmits an Echo Request that has a Routing Header to an IPv6 destination address of the RUT's Solicited-Node Multicast Address. The Echo Request is destined for TN1 with the RUT as the first hop.
E. *Hop Limit = 0*: TN2 transmits an Echo Request that has a Routing Header with a Hop Limit field of value 0 in the IPv6 header. The Echo Request is destined for TN1 with the RUT as the first hop.
F. *Hop Limit = 1*: TN2 transmits an Echo Request that has a Routing Header with a Hop Limit field of value 1 in the IPv6 header. The Echo Request is destined for TN1 with the RUT as the first hop.
G. *Valid Processing*: TN2 transmits an Echo Request that has a Routing header with three segments, the first of which is through TR1. The Echo Request is destined for TN1 with the RUT as the first hop.



**Comments on Test Results**

*Observations of Setup for Each Part:* The RUT Solicited TR1 and sent an Echo Reply to TR1, no other packets were sent, indicating the RUT's NCE for TR1 is in state **REACHABLE**.

A. The RUT discarded the Echo Request and sent an ICMPv6 Parameter Problem message to TN2's Global Address. The Parameter Problem message had a code field of 0, and an invalid Pointer field of 0x01.

According to RFC 2460, Section 4.4: "If the header extension length is odd, send an ICMP Parameter Problem message, Code 0, to the Source Address, pointing to the Hdr Ext Len field, and discard the packet."

Therefore, the RUT should have sent to TN2's Global Address an ICMPv6 Parameter Problem message, Code 0, with a Pointer field of 0x29 (offset of the Header Extension Length field of the Routing Header).

B. The RUT discarded the Echo Request and sent an ICMPv6 Parameter Problem message to TN2's Global Address. The Parameter Problem message had a code field of 0, and a Pointer field of 0x2B (offset of the Segments Left field of the Routing Header).
C. The RUT discarded the Echo Request.
D. The RUT discarded the Echo Request.

E. The RUT discarded the Echo Request and transmitted a Destination Unreachable message to TN2's Global Address.

According to RFC2460, Section 4.4: "If the Ipv6 Hop Limit is less than or equal to 1, send an ICMP Time Exceeded – Hop Limit Exceeded in Transit message to the Source address and discard the packet"

Therefore, the RUT should have discarded the Echo Request from TN2 and transmitted an ICMPv6 Time Exceeded – Hop Limit Exceeded in Transit Message to TN2's Global Address.

F. The RUT discarded the Echo Request and transmitted a Destination Unreachable message to TN2's Global Address.

According to RFC2460, Section 4.4: "If the Ipv6 Hop Limit is less than or equal to 1, send an ICMP Time Exceeded – Hop Limit Exceeded in Transit message to the Source address and discard the packet"

Therefore, the RUT should have discarded the Echo Request from TN2 and transmitted an ICMPv6 Time Exceeded – Hop Limit Exceeded in Transit Message to TN2's Global Address.

G. The RUT transmitted the Echo Request to TR1 with the following values:

- IPv6 Source Address TN2's Global Address
- IPv6 Destination Address TR1's Global Address
- Hop Limit 9

Routing Header:

- Hdr. Ext. Length 6
- Routing Type 0
- Segments Left 2
- Address[1] RUT's Global Address
- Address[2] Global Address 3
- Address[3] TN1's Global Address

## Group 3: Fragmentation

The following tests cover fragmentation in IPv6.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.3.1 | **Fragment Reassembly** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |

**Purpose**: Verify that a node correctly reassembles fragmented packets and distinguishes between packet fragments using the Source Address, Destination Address, and Fragment ID.

**Comments on Test Procedure**

A. *All Fragments are Valid*: TN1 transmits 3 Fragments. Each has the same Source Address, Destination Address, and Fragment ID. The fragments come in normal order with the correct fragment offsets.
B. *All Fragments are Valid, reverse order*: TN1 transmits 3 Fragments. Each has the same Source Address, Destination Address, and Fragment ID. The fragments come in reverse order with the correct fragment offsets.
C. *Fragment Ids Differ Between Fragments*: TN1 transmits 3 Fragments. The fragments come in normal order with the correct fragment offsets. The first and last fragments have a Fragment ID of 2999, while the second has a Fragment ID of 3000. The Source and Destination Address of each packet is the same.
D. *Source Addresses Differ Between Fragments*: TN1 transmits 3 Fragments. The fragments come in normal order with the correct fragment offsets. The first and last fragments have a Source Address of the Link-Local address of TN1, while the second has a different Source Address. The Destination Address and Fragment IDs of each packet is the same.
E. *Destination Address Differ Between Fragments*: TN1 transmits 3 Fragments. The fragments come in normal order with the correct fragment offsets. The first and last fragments have a Destination Address of the Link-Local address of TN1, while the second has a different Source Address. The Source Address and Fragment IDs of each packet is the same.
F. *Reassemble to 1500*: TN1 transmits 3 Fragments. Each has the same Source Address, Destination Address, and Fragment ID. The fragments come in normal order with the correct fragment offsets, however, the payloads of each fragment are modified so that the reassembled packet size is 1500.

**Comments on Test Results**

A. The NUT transmitted an Echo Reply to TN1 in response to the reassembled Echo Request.
B. The NUT transmitted an Echo Reply to TN1 in response to the reassembled Echo Request.
C. The NUT did not transmit an Echo Reply to TN1, due to the Echo Request not being reassembled because of different Fragment IDs. The NUT transmitted an ICMPv6 Time Exceeded Message to TN1 sixty seconds after it received the first fragment.
D. The NUT did not transmit an Echo Reply to TN1, due to the Echo Request not being reassembled because of different Source Addresses. The NUT transmitted an ICMPv6 Time Exceeded Message to TN1 sixty seconds after it received the first fragment.
E. The NUT did not transmit an Echo Reply to TN1, due to the Echo Request not being reassembled because of different Destination Addresses. The NUT transmitted an ICMPv6 Time Exceeded Message to TN1 sixty seconds after it received the first fragment.
F. The NUT responded to the Echo Request from TN1.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.3.2 | **Reassembly Time Exceeded** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |

**Purpose**: Verify that a node takes the proper actions when the reassembly time has been exceeded for a packet.

**Comments on Test Procedure**

A. *Time Elapsed Between Fragments less than Sixty Seconds*: TN1 transmits three Fragments, in normal order. There is a 55-second delay between the transmission of the first and second fragments.
B. *Time Exceeded Before Last Fragments Arrive*: TN1 transmits three Fragments, in normal order. There is a 65-second delay between the transmission of the first and second fragments.
C. *Time Exceeded, Only First Fragment Received*: TN1 transmits only the first of many fragments.
D. *Time Exceeded, Only Second Fragment Received*: TN1 transmits only the second of many fragments.

**Comments on Test Results**

A. The NUT's reassembly timer expired just after it received the second and third fragments, and transmitted an Echo Reply to TN1 in response to the reassembled Echo Request.
B. The NUT's reassembly timer expired just before it received the second and third fragments, and did not transmit an Echo Reply to TN1 since the Echo Request could not be reassembled in time. The NUT transmitted an ECMPv6 Time Exceeded Message to TN1 sixty seconds after it received the first fragment, with the following properties:
    - The Code Field has a value of 1 (Fragment Reassembly Time Exceeded)
    - The unused field was initialized to zero
    - The Source Address of the Packet was the same as the Global Destination Address of TN1's Echo Request Packet
    - The Destination Address was the same as the Global Source Address of TN1's Echo Request Packet
    - The size of the invoking Echo Request packet included in the Error Message did not exceed minimum IPv6 MTU.
C. The NUT's reassembly timer expired just before it received the second and third fragments, and did not transmit an Echo Reply to TN1 since the Echo Request could not be reassembled in time. The NUT transmitted an ECMPv6 Time Exceeded Message to TN1 sixty seconds after it received the first fragment, with the following properties:
    - The Code Field has a value of 1 (Fragment Reassembly Time Exceeded)
    - The unused field was initialized to zero
    - The Source Address of the Packet was the same as the Global Destination Address of TN1's Echo Request Packet
    - The Destination Address was the same as the Global Source Address of TN1's Echo Request Packet
    - The size of the invoking Echo Request packet included in the Error Message did not exceed minimum IPv6 MTU.
D. The NUT did not transmit an Echo Reply or an ICMPv6 Time Exceed Message to TN1.

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.3.3 | **Fragment Header M-Bit Set, Payload Length Invalid** | **A** | **PASS** |
| **Purpose**: Verify that a node takes the proper actions when it receives a fragment with the M-bit set (more fragments), but which has a Payload Length that is not a multiple of 8 bytes. | | | |
| **Comments on Test Procedure** | | | |
| A. TN1 transmits an Echo Request that has a Fragment Header with the M-bit set. The Payload length is 21, which is not a multiple of 8 octets. | | | |
| **Comments on Test Results** | | | |
| A. The NUT did not transmit an Echo Reply to TN1, since the fragment was discarded. The NUT transmitted an ICMPv6 Parameter Problem message to TN1 with a Code Field of 0 (erroneous header field encounter), and a Pointer field of 0x04 (offset of the Payload Length field of the IPv6 header). | | | |

| Test # | | Result | |
|---|---|---|---|
| v6LC.1.3.4 | **Stub Fragment Header** | **A** | **PASS** |
| **Purpose**: Verify that a node accepts the offset zero fragment with the More Fragments flag clear. | | | |
| **Comments on Test Procedure** | | | |
| A. TN1 transmits an Echo Request that has a Fragment Header with a Fragment Offset of 0 and the More Fragments flag clear. | | | |
| **Comments on Test Results** | | | |
| A. The NUT transmitted an Echo Reply, which did not include a fragment header, to TN1. | | | |

# Section 2: RFC 2461

These tests are designed to verify the readiness of an IPv6 implementation vis-à-vis the Neighbor Discovery specification.

## Group 1: Address Resolution and Neighbor Unreachability Detection

The following tests cover Address Resolution and Neighbor Unreachability Detection in IPv6.



| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.1 | **On-link Determination** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |

**Purpose**: Verify that a node correctly determines that a destination is on-link.

**Comments on Test Procedure**

A. *Link-local Address*: TN1 transmits an Echo Request with TN1's Link-Local source address.
B. *Global Address, No Default Router*: TN1 transmits an Echo Request with TN1's Global Address.
C. *Global Address, On-link Prefix covers TN1*: TR1 transmits a Router Advertisement with a Prefix Length of 64, the "L-Bit" set (on-link) and prefix of TN1's Global Prefix.
D. *Global Address, On-link Prefix does not cover TN2*: TR1 transmits a Router Advertisement with a Prefix Length of 64, the "L-Bit" set (on-link) and prefix of TN1's Global Prefix. The Prefix Advertisement does not cover TN2's Global Address. TN2 transmits an Echo Request with TN2's Global Address.

**Comments on Test Results**

A. The NUT sent a Neighbor Solicitation with a Target Address equal to TN1's Link-Local address, an indication that the NUT has successfully determined that TN1 was on-link.
B. Since there were no routers on the link, the NUT considered all global addresses as being on-link. Thus, the NUT sent a Neighbor Solicitation with a Target Address equal to TN1's Global Address, an indication that the NUT has successfully determined that TN1 was on-link.
C. TN1's Global Address was covered by the on-link prefix. Hence, the NUT considered TN1's Global Address as on-link. The NUT sent a Neighbor Solicitation with a Target Address equal to TN1's Global Address, an indication that the NUT successfully determined that TN1 was on-link.
D. TN2's Global Address was not covered by the on-link prefix. Hence, the NUT considered TN2's Global Address as off-link. The NUT sent a Neighbor Solicitation with a Target Address equal to TR1's address, an indication that the NUT successfully determined that TN2 was off-link.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.2 | **Resolution Wait Queue** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a node properly queues packets while waiting for address resolution of the next hop.

**Comments on Test Procedure**

A. *Single* Queue: TN1 transmits an Echo Request 3 times, each time incrementing its ID. TN1 then transmits a Neighbor Advertisement with the Router flag not set, and the Solicited and Override flags set, and a Target Address of TN1's Link-Local Address, in response to any Neighbor Solicitations from the NUT.
B. *Multiple Queues*: TN1 transmits an Echo Request 3 times, each time incrementing its ID, followed by another group of 4 Echo Requests, with increasing ID numbers. TN1 and TN2 transmit Neighbor Advertisements with the Router flag not set, and the Solicited and Override flags set, and a Target Address of either TN1's Link-Local Address or TN2's Link-Local Address, in response to any Neighbor Solicitations from the NUT.

**Comments on Test Results**

A. The NUT transmitted a Neighbor Solicitation with a Target Address equal to TN1's Link-Local address, followed by Echo Replies in response to the 3 Requests. The Echo Replies corresponded to the last 3 Echo Requests sent by TN1 to the NUT, an indication that the queuing of packets while waiting for address resolution to complete was done successfully. The number of Echo Replies was not less than 1.
B. The NUT transmitted a Neighbor Solicitation with a Target Address equal to TN1's Link-Local address, followed by Echo Replies in response to the 3 Requests. The Echo Replies corresponded to the last 3 Echo Requests sent by TN1 to the NUT, an indication that the queuing of packets while waiting for address resolution to complete was done successfully. The number of Echo Replies was not less than 1. The NUT transmitted a Neighbor Solicitation with a Target Address equal to TN2's Link-Local address, followed by Echo Replies in response to the 4 Requests. The Echo Replies corresponded to the last 3 Echo Requests sent by TN2 to the NUT, an indication that the queuing of packets while waiting for address resolution to complete was done successfully. The number of Echo Replies was not less than 1.



| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.3 | **Prefix Information Option Processing, On-link Flag (Hosts Only)** | **A** | **N/A** |

**Purpose**: Verify that a host properly processes the on-link flag of a Prefix Information Option.

**Comments on Test Procedure**

A. TR1 transmits a Router Advertisement with a Lifetime of 100 seconds, Reachable time of 10 seconds, a Retrans Interval of 1 second, with a Valid and Preferred Prefix Lifetime of 20 seconds a prefix of TR1's Global Prefix and the on-link flag set. TR1 then transmits an Echo Request. TR1 does not respond to Neighbor Solicitations from the HUT. TR1 transmits the same Router Advertisement, but with the on-link flag clear. TR1 then transmits an Echo Request , and does not respond to Neighbor Solicitation from the HUT.

**Comments on Test Results**

A. This test is performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.4 | **Prefix Invalidation (Hosts Only)** | **A** | **N/A** |

**Purpose**: Verify that a host takes the appropriate actions when the Invalidation Timer expires for a Prefix List entry.

**Comments on Test Procedure**

A. TR1 transmits a Router Advertisement with a Lifetime of 100 seconds, Reachable time of 10 seconds, a Retrans interval of 1 second, with a Valid and Preferred Prefix Lifetime of 10 seconds a Prefix of TR1's Global Prefix and the on-link flag set. TR1 then transmits an Echo Request to the HUT's Link-Local address. TR1 transmits a Neighbor Advertisement with the Router Flag clear, and the Solicited and Override flags set, in response to any Neighbor Solicitations from the HUT. TR1 transmits the same Router Advertisement as previously, but with the Preferred and Valid Lifetime fields set to 0. TR1 then transmits an Echo Request to the HUT's Link-Local address.

**Comments on Test Results**

A. This test is performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.5 | **Host Prefix List (Hosts Only)** | **A-B** | **N/A** |

**Purpose**: Verify that a host properly updates its Prefix List upon receipt of Prefix Information Options, which have the on-link flag set.

**Comments on Test Procedure**

A. *Prefix Lifetime has not Expired*: TR1 transmits a Router Advertisement with a Lifetime of 20 seconds, a Reachable time of 600 seconds and a Retrans time of 1 second, and no Prefix option. TR1 transmits a link-local Echo Request to the HUT, and transmits a Neighbor Advertisement in response to any Neighbor Solicitations from the HUT. TR1 then transmits the same Router Advertisement, but with the Prefix option header, the on-link flag set, Valid and Preferred Lifetimes of 10 seconds, and a prefix of TN1's Global Prefix. After 8 seconds TN1 transmits an Echo Request to the NUT's Link-Local address, the source address is covered by the prefix advertised in the Router Advertisement.

B. *Prefix Lifetime updated by Router Advertisement*: TR1 transmits a Router Advertisement with a Lifetime of 20 seconds, a Reachable time of 600 seconds and a Retrans time of 1 second, and no prefix option. TR1 transmits a link-local Echo Request to the HUT, and transmits a Neighbor Advertisement in response to any Neighbor Solicitations from the HUT. TR1 transmits the same Router Advertisement, but with the prefix option header, the on-link flag set, Valid and Preferred Lifetimes of 10 seconds, and a prefix of TN1's Global Prefix. After 8 seconds, retransmit the Routing Advertisement. After 8 more seconds, TN1 transmits an Echo Request to the NUT's Link-Local address, the source address is covered by the prefix advertised in the Router Advertisement.

**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.6 | **Neighbor Solicitation Origination, Address Resolution** | **A** | **FAIL** |
| | | **B** | **FAIL** |

**Purpose**: Verify that a node properly originates Neighbor Solicitations when trying to resolve the address of a neighbor.

**Comments on Test Procedure**

A. *Neighbor Solicitation Origination, Target Address Being Link-Local*: If the NUT is a host, perform Common Test Setup 1.1 with a Retransmit Interval value of 1 second. If the NUT is a router, configure the Retransmit Interval value to 1 second. TR1 transmits an Echo Request to the NUT and responds to Neighbor Solicitations from the NUT. Wait for an Echo Reply from the NUT. This should cause the NUT to resolve the address of TR1 and create a Neighbor Cache entry for TR1 in state **REACHABLE**. TN1 transmits an Echo Request (Packet A) from its Link-Local address to the NUT's Link-Local address. TN1 does not send any Neighbor Advertisements. Repeat with a Retransmit Interval value of 5 seconds.

B. *Neighbor Solicitation Origination, Target Address Being Global*: If the NUT is a host, perform Common Test Setup 1.1 with a Retransmit Interval value of 1 second. If the NUT is a router, configure the Retransmit Interval value to 1 second. TR1 transmits an Echo Request to the NUT and responds to Neighbor Solicitations from the NUT. Wait for an Echo Reply from the NUT. This should cause the NUT to resolve the address of TR1 and create a Neighbor Cache entry for TR1 in state **REACHABLE**. TN1 transmits an Echo Request (Packet A) from its Global Address to the NUT's Global Address. TN1 does not send any Neighbor Advertisements. Repeat with a Retransmit Interval value of 5 seconds.

**Comments on Test Results**

A. In response to Packet A (referenced above), the NUT transmitted Neighbor Solicitations with a Target Address equal to TN1's Link-Local Address at intervals of 1 second. The NUT transmitted no more than 1 Neighbor Solicitation per second. Each Neighbor Solicitation had a Source Link-Layer Address Option. The maximum number of Neighbor Solicitations was 3, which is equal to MAX_MULTICAST_SOLICIT. This test could not be repeated with a Retransmit value of 5 seconds because that variable was unable to be configured by System Management, and the external behavior of the device did not change.

According to RFC 2461, Section 6.2.1: "A router MUST allow for [Neighbor Solicitation Retransmit Interval] to be configured by system management. […] an implementation is not required to have [it] so long as its external behavior is consistent with that described in this document. […] Some of these host variables (e.g., CurHopLimit, RetransTimer, and ReachableTime) apply to all nodes including routers."

Therefore, the NUT should have allowed for the configuration of the Neighbor Solicitation Retransmit Interval to 5 seconds so that it transmitted Neighbor Solicitations with a Target Address equal to TN1's Link-Local Address at intervals of 5 seconds.

B. In response to Packet A (referenced above), the NUT transmitted Neighbor Solicitations with a Target Address equal to TN1's Link-Local Address at intervals of 1 second. The NUT transmitted no more than 1 Neighbor Solicitation per second. Each Neighbor Solicitation had a Source Link-Layer Address Option. The maximum number of Neighbor Solicitations was 3, which is equal to MAX_MULTICAST_SOLICIT. This test could not be repeated with a Retransmit value of 5 seconds because that variable was unable to be configured by System Management, and the external behavior of the device did not change.

According to RFC 2461, Section 6.2.1: "A router MUST allow for [Neighbor Solicitation Retransmit Interval] to be configured by system management. […] an implementation is not required to have [it] so long as its external behavior is consistent with that described in this document. […] Some of these host variables (e.g., CurHopLimit, RetransTimer, and ReachableTime) apply to all nodes including routers."

Therefore, the NUT should have allowed for the configuration of the Neighbor Solicitation Retransmit Interval to 5 seconds so that it transmitted Neighbor Solicitations with a Target Address equal to TN1's Global Address at intervals of 5 seconds.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.7 | **Neighbor Solicitation Origination, Reachability Confirmation** | **A** | **PASS** |

**Purpose**: Verify that a node properly originates Neighbor Solicitations when trying to confirm the reachability of a neighbor.

**Comments on Test Procedure**

A. If the NUT is a host, TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. This should cause the NUT to add TR1 to its Default Router List, configure a global address, and compute Reachable Time. The Router and Prefix Lifetimes are long enough such that they do not expire during the test. The Retransmit Interval value is 1 second. If the NUT is a router, configure a default route with TR1 as the next hop and configure Base Reachable Time for the Router to 30 seconds. TR1 transmits an Echo Request to the NUT and responds to Neighbor Solicitations from the NUT. Wait for an Echo Reply from the NUT. This should cause the NUT to resolve the address of TR1 and create a Neighbor Cache entry for TR1 in state **REACHABLE**. TN1 is to transmit an Echo Request (Packet A) from its Link-Local address to the NUT's Link-Local address. TN1 is to send a Neighbor Advertisement upon receiving Neighbor Solicitations from the NUT. Wait (REACHABLE_TIME * MAX_RANDOM_FACTOR) seconds so that the NCE of TN1 transit to state STALE and then transmit an Echo Request (Packet A) from TN1 Link-Local address to the NUT's Link-Local address.

**Comments on Test Results**

A. In response to Packet A, the NUT transmitted Neighbor Solicitations with a Target Address equal to the TN1's Link-Local Address at intervals of 1 second. The NUT then transmitted 1 Neighbor Solicitation every 1 second. Once the Neighbor Advertisement was received from TN1, the NUT sent an Echo Reply in response to Packet A. The NCE of TN1 was set to state **REACHABLE**. In response to Packet A, the NUT transmitted an Echo Reply. After (DELAY_FIRST_PROBE_TIME) the NUT transmitted Neighbor Solicitations with the NUT's Link-Local address being the source address and TN1's Link-Local address as the destination address. The maximum number of Neighbor Solicitations that the NUT can transmit is 3.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.8 | **Invalid Neighbor Solicitation Handling** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |
| | | **G** | **PASS** |
| | | **H** | **PASS** |

**Purpose**: Verify that a node takes the proper actions upon receipt of an invalid Neighbor Solicitation.

**Comments on Test Procedure**

A. *Invalid Target Address:* TN1 transmits a Neighbor Solicitation from its Link-Local Address to the NUT's Link-Local Address the Target Address is set to the All Nodes Multicast.
B. *Invalid Destination Address:* TN1 transmits a Neighbor Solicitation with a Source Address of the Unspecified Address (::), Destination Address of the NUT's Link-Local Address and a Target Address of the NUT's Link-local Address.
C. *Invalid Source Link-layer Address Option:* TN1 transmits a Neighbor Solicitation with a Source Address equal to the Unspecified Address (::) and a Destination of the Solicited-Node Multicast Address of the ?????.
D. *Invalid Hop Limit:* TN1 transmits a Neighbor Solicitation (Packet A) with the Hop Limit set to 254.
E. *Invalid Checksum:* TN1 transmits a Neighbor Solicitation (Packet A) with the ICMP checksum set to be invalid.
F. *Invalid ICMP code:* TN1 transmits a Neighbor Solicitation (Packet A) with the ICMP Code set to 1.
G. *Invalid ICMP Length:* TN1 transmits a Neighbor Solicitation (Packet A) with the ICMP Length set to 16.
H. *Option of Length 0*: TN1 transmits a Neighbor Solicitation (Packet A) with an Option Length set to 0.

**Comments on Test Results**

A-H. The NUT did not transmit any packets corresponding to the Neighbor Solicitation.



| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.9 | **Neighbor Solicitation Processing, No NCE** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a node properly updates its neighbor cache upon receipt of neighbor solicitations when there is no NCE exists for that neighbor.

**Comments on Test Procedure**

A. *Unicast Neighbor Solicitation:* TN1 transmits a Neighbor Solicitation (Packet A).
B. *Multicast Neighbor Solicitation*: TN1 transmits a Neighbor Solicitation (Packet B).

**Comments on Test Results**

A. The NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to STALE. The NUT replied to the Neighbor Solicitation (Packet A) by sending a Neighbor Advertisement and set the state of the Entry to DELAY. After DELAY_FIRST_PROBE_TIME, the NUT sent a Unicast Neighbor Solicitation to TN1.
B. The NUT should create a Neighbor Cache Entry for TN1 and set the state of the Entry to STALE. The NUT should reply to Neighbor Solicitation B by sending a Neighbor Advertisement and set the state of the Entry to DELAY. After DELAY_FIRST_PROBE_TIME, the NUT should send a Unicast Neighbor Solicitation to TN1.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.10 | **Neighbor Solicitation Processing, NCE State INCOMPLETE** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a node properly updates its neighbor cache upon receipt of neighbor solicitations when the NCE of the neighbor is in state INCOMPLETE.

**Comments on Test Procedure**

A. *Unicast Neighbor Solicitation:* TN1 transmits an Echo Request (Packet A) and then TN1 transmits a Neighbor Solicitation (Packet B).
B. *Multicast Neighbor Solicitation:* TN1 transmits an Echo Request (Packet A) and then TN1 transmits a Neighbor Solicitation (Packet C).

**Comments on Test Results**

A. The NUT created a Neighbor Cache Entry for TN1. TN1 set the state of the Entry to INCOMPLETE. The NUT sent a multicast Neighbor Solicitation to TN1 which after receiving TN1's Neighbor Solicitation, the NUT sent its queued Echo Reply to TN1. The NUT then updated the NCE of TN1 to state STALE and updated its link-layer address for TN1 accordingly. The NUT replied to the Neighbor Solicitation (Packet B) by sending a Neighbor Advertisement and set the state of the Entry to DELAY. After DELAY_FIRST_PROBE_TIME, the NUT sent a Unicast Neighbor Solicitation to TN1.
B. The NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to INCOMPLETE. The NUT sent a multicast Neighbor Solicitation to TN1 which after receiving TN1's Neighbor Solicitation, the NUT sent its queued Echo Reply to TN1. The NUT then updated the NCE of TN1 to state STALE and updated its link-layer address for TN1 accordingly. The NUT replied to the Neighbor Solicitation (Packet C) by sending a Neighbor Advertisement and set the state of the Entry to DELAY. After DELAY_FIRST_PROBE_TIME, the NUT sent a Unicast Neighbor Solicitation to TN1.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.11 | **Neighbor Solicitation Processing, NCE State REACHABLE** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |

**Purpose**: Verify that a node properly updates its neighbor cache upon receipt of neighbor solicitations when the NCE of the neighbor is in state **REACHABLE**.

**Comments on Test Procedure**

A. *Unicast Neighbor Solicitation with the same SLLA:* TN1 transmits an Echo Request (Packet A) and then TN1 transmits a solicited Neighbor Advertisement (Packet B). The NCE should be set to **REACHABLE** after TN1 sends out the Echo Reply. TN1 transmits another Echo Request (Packet A) and TN1 transmits a Neighbor Solicitation (Packet C).

B. *Unicast Neighbor Solicitation with a different SLLA:* TN1 transmits an Echo Request (Packet A) and then TN1 transmits a solicited Neighbor Advertisement (Packet B). The NCE should be sent to **REACHABLE** after TN1 sends out the Echo Reply. TN1 transmits another Echo Request (Packet A) and TN1 transmits a Neighbor Solicitation (Packet C) with a different address as the Source Link-layer Address.

C. *Multicast Neighbor Solicitation with the same SLLA*: TN1 transmits an Echo Request (Packet A) and then TN1 transmits a solicited Neighbor Advertisement (Packet B). The NCE should be sent to **REACHABLE** after TN1 sends out the Echo Reply. TN1 transmits another Echo Request (Packet A) and TN1 transmits a Neighbor Solicitation (Packet D).

D. *Multicast Neighbor Solicitation with a different SLLA*: TN1 transmits an Echo Request (Packet A) and then TN1 transmits a solicited Neighbor Advertisement (Packet B). The NCE should be sent to **REACHABLE** after TN1 sends out the Echo Reply. TN1 transmits another Echo Request (Packet A) and TN1 transmits a Neighbor Solicitation (Packet D) with a different address as the Source Link-layer Address.



**Comments on Test Results**

A. The NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to INCOMPLETE. The NUT sent a multicast Neighbor Solicitation to TN1, which after receiving TN1's Neighbor Advertisement, the NUT should send its queued Echo Reply to TN1. The NUT then updated the NCE of TN1 to state **REACHABLE** and updated its link-layer address for TN1 accordingly. Because the NUT is in state **REACHABLE**, after receiving the Echo Request from TN1, the NUT sent an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT did not send a unicast Neighbor Solicitation to TN1. The NUT did not update the NCE of TN1, the NUT replied to the Neighbor Solicitation (Packet C) by sending a Neighbor Advertisement. TN1 stayed in state **REACHABLE**.

B. The NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to INCOMPLETE. The NUT sent a multicast Neighbor Solicitation to TN1, which after receiving TN1's Neighbor Advertisement, the NUT should send its queued Echo Reply to TN1. The NUT then updated the NCE of TN1 to state **REACHABLE** and updated its link-layer address for TN1 accordingly. Because the NUT is in state **REACHABLE**, after receiving the Echo Request from TN1, the NUT sent an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT did not send a unicast Neighbor Solicitation to TN1. The NUT updated the NCE of TN1 to state STALE and updated TN1's Link-layer address to its new Link-layer address from the received Neighbor Solicitation (Packet C). The NUT replied to Neighbor Solicitation (Packet C) by sending a Neighbor Advertisement and set the state of the Entry to DELAY. After DELAY_FIRST_PROBE_TIME, the NUT sent a unicast Neighbor Solicitation to TN1 with the Target set the new Link-Layer address of TN1.

C. The NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to INCOMPLETE. The NUT sent a multicast Neighbor Solicitation to TN1, which after receiving TN1's Neighbor Advertisement, the NUT should send its queued Echo Reply to TN1. The NUT then updated the NCE of TN1 to state **REACHABLE** and updated its link-layer address for TN1 accordingly. Because the NUT is in state **REACHABLE**, after receiving the Echo Request from TN1, the NUT sent an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT did not send a unicast Neighbor Solicitation to TN1. The NUT did not update the NCE of TN1, the NUT

replied to the Neighbor Solicitation (Packet D) by sending a Neighbor Advertisement. TN1 stayed in state **REACHABLE**.

D. The NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to INCOMPLETE. The NUT sent a multicast Neighbor Solicitation to TN1, which after receiving TN1's Neighbor Advertisement, the NUT should send its queued Echo Reply to TN1. The NUT then updated the NCE of TN1 to state **REACHABLE** and updated its link-layer address for TN1 accordingly. Because the NUT is in state **REACHABLE**, after receiving the Echo Request from TN1, the NUT sent an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT did not send a unicast Neighbor Solicitation to TN1. The NUT updated the NCE of TN1 to state STALE and updated TN1's Link-layer address to its new Link-layer address from the received Neighbor Solicitation (Packet C). The NUT replied to Neighbor Solicitation (Packet C) by sending a Neighbor Advertisement and set the state of the Entry to DELAY. After DELAY_FIRST_PROBE_TIME, the NUT sent a unicast Neighbor Solicitation to TN1 with the Target set the new Link-Layer address of TN1.



| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.12 | **Neighbor Solicitation Processing, NCE State STALE** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |

**Purpose**: Verify that a node properly updates its neighbor cache upon receipt of neighbor solicitations when the NCE of the neighbor is in state STALE.

**Comments on Test Procedure**

A. *Unicast Neighbor Solicitation with the same SLLA:* TN1 transmits an Echo Request (Packet A) and then TN1 transmits a solicited Neighbor Advertisement (Packet B). The NCE should be set to **REACHABLE** after TN1 sends out the Echo Reply. TN1 transmits another Echo Request (Packet A) and waits (REACHABLE_TIME * MAX_RANDOM_FACTOR) seconds to transmit a Neighbor Solicitation (Packet C).

B. *Unicast Neighbor Solicitation with a different SLLA:* TN1 transmits an Echo Request (Packet A) and then TN1 transmits a solicited Neighbor Advertisement (Packet B). The NCE should be set to **REACHABLE** after TN1 sends out the Echo Reply. TN1 transmits another Echo Request (Packet A) and waits (REACHABLE_TIME * MAX_RANDOM_FACTOR) seconds to transmit a Neighbor Solicitation with a different address as the Source Link-layer Address (Packet C).

C. *Multicast Neighbor Solicitation with the same SLLA:* TN1 transmits an Echo Request (Packet A) and then TN1 transmits a solicited Neighbor Advertisement (Packet B). The NCE should be sent to **REACHABLE** after TN1 sends out the Echo Reply. TN1 transmits another Echo Request (Packet A) and waits (REACHABLE_TIME * MAX_RANDOM_FACTOR) seconds to transmit a Neighbor Solicitation (Packet D).

D. *Multicast Neighbor Solicitation with a different SLLA*: TN1 transmits an Echo Request (Packet A) and then TN1 transmits a solicited Neighbor Advertisement (Packet B). The NCE should be sent to **REACHABLE** after TN1 sends out the Echo Reply. TN1 transmits another Echo Request (Packet A) and waits (REACHABLE_TIME * MAX_RANDOM_FACTOR) seconds to transmit a Neighbor Solicitation with a different address as the Source Link-layer Address (Packet D).

**Comments on Test Results**

A. The NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to INCOMPLETE. The NUT sent a multicast Neighbor Solicitation to TN1, which after receiving TN1's Neighbor Advertisement, the NUT should send its queued Echo Reply to TN1. The NUT then updated the NCE of TN1 to state **REACHABLE**

and updated its link-layer address for TN1 accordingly. Because the NUT is in state **REACHABLE**, after receiving the Echo Request from TN1, the NUT sent an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT did not send a Unicast Neighbor Solicitation to TN1. The NUT should update the NCE of TN1 to state STALE, and after transmitting the Neighbor Solicitation (Packet C), the NUT did not update its NCE and the NUT stayed in state STALE. The NUT replied to the Neighbor Solicitation by sending a Neighbor Advertisement and set the state of the TN1's Entry to DELAY. After DELAY_FIRST_PROBE_TIME, the NUT sent a unicast Neighbor Solicitation to TN1.

B. The NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to INCOMPLETE. The NUT sent a multicast Neighbor Solicitation to TN1, which after receiving TN1's Neighbor Advertisement, the NUT should send its queued Echo Reply to TN1. The NUT then updated the NCE of TN1 to state **REACHABLE** and updated its link-layer address for TN1 accordingly. Because the NUT is in state **REACHABLE**, after receiving the Echo Request from TN1, the NUT sent an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT did not send a Unicast Neighbor Solicitation to TN1. The NUT should update the NCE of TN1 to state STALE, and the NUT should update TN1's Link-Layer address to its new Link-Layer address from the received Neighbor Solicitation (Packet C). The NUT did not update the NCE of TN1 and stayed in state STALE. The NUT replied to the Neighbor Solicitation by sending a Neighbor Advertisement to TN1's new Link-Layer address and set the state of the TN1's Entry to DELAY. After DELAY_FIRST_PROBE_TIME, the NUT sent a unicast Neighbor Solicitation to TN1 using the new link-layer address as the Target.

C. The NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to INCOMPLETE. The NUT sent a multicast Neighbor Solicitation to TN1, which after receiving TN1's Neighbor Advertisement, the NUT should send its queued Echo Reply to TN1. The NUT then updated the NCE of TN1 to state **REACHABLE** and updated its link-layer address for TN1 accordingly. Because the NUT is in state **REACHABLE**, after receiving the Echo Request from TN1, the NUT sent an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT did not send a Unicast Neighbor Solicitation to TN1. The NUT should update the NCE of TN1 to state STALE, and after transmitting the Neighbor Solicitation (Packet D), the NUT did not update its NCE and the NUT stayed in state STALE. The NUT replied to the Neighbor Solicitation by sending a Neighbor Advertisement and set the state of the TN1's Entry to DELAY. After DELAY_FIRST_PROBE_TIME, the NUT sent a unicast Neighbor Solicitation to TN1.

D. The NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to INCOMPLETE. The NUT sent a multicast Neighbor Solicitation to TN1, which after receiving TN1's Neighbor Advertisement, the NUT should send its queued Echo Reply to TN1. The NUT then updated the NCE of TN1 to state **REACHABLE** and updated its link-layer address for TN1 accordingly. Because the NUT is in state **REACHABLE**, after receiving the Echo Request from TN1, the NUT sent an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT did not send a Unicast Neighbor Solicitation to TN1. The NUT should update the NCE of TN1 to state STALE, and the NUT should update TN1's Link-Layer address to its new Link-Layer address from the received Neighbor Solicitation (Packet D). The NUT did not update the NCE of TN1 and stayed in state STALE. The NUT replied to the Neighbor Solicitation by sending a Neighbor Advertisement to TN1's new Link-Layer address and set the state of the TN1's Entry to DELAY. After DELAY_FIRST_PROBE_TIME, the NUT sent a unicast Neighbor Solicitation to TN1 using the new link-layer address as the Target.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.13 | **Neighbor Solicitation Processing, NCE State PROBE** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |

**Purpose**: Verify that a node properly updates its neighbor cache upon receipt of neighbor solicitations when the NCE of the neighbor is in state Probe.

**Comments on Test Procedure**

A. *Unicast Neighbor Solicitation with the same SLLA:* TN1 transmits an Echo Request (Packet A) to the NUT and also transmits a Neighbor Advertisement (Packet B) to the NUT after receiving any Neighbor Solicitations from the NUT. The NUT should update the NCE of TN1 to state STALE and waits (DELAY_FIRST_PROBE_TIME) seconds to transmit a Neighbor Solicitation (Packet C).
B. *Unicast Neighbor Solicitation with a different SLLA:* TN1 transmits an Echo Request (Packet A) to the NUT and also transmits a Neighbor Advertisement (Packet B) to the NUT after receiving any Neighbor Solicitations from the NUT. The NUT should update the NCE of TN1 to state STALE and waits (DELAY_FIRST_PROBE_TIME) seconds to transmit a Neighbor Solicitation (Packet C) with a different address as the Source Link-Layer Address.
C. *Multicast Neighbor Solicitation with the same SLLA:* TN1 transmits an Echo Request (Packet A) to the NUT and also transmits a Neighbor Advertisement (Packet B) to the NUT after receiving any Neighbor Solicitations from the NUT. The NUT should update the NCE of TN1 to state STALE and waits (DELAY_FIRST_PROBE_TIME) seconds to transmit a Neighbor Solicitation (Packet D).
D. *Multicast Neighbor Solicitation with a different SLLA:* TN1 transmits an Echo Request (Packet A) to the NUT and also transmits a Neighbor Advertisement (Packet B) to the NUT after receiving any Neighbor Solicitations from the NUT. The NUT should update the NCE of TN1 to state STALE and waits (DELAY_FIRST_PROBE_TIME) seconds to transmit a Neighbor Solicitation (Packet D) with a different address as the Source Link-Layer Address.



**Comments on Test Results**

A. The NUT updated the NCE of TN1 to state STALE. After receiving the Echo Request from TN1, the NUT sent an Echo Reply and entered state DELAY. After DELAY_FIRST_PROBE_TIME, the NUT transitioned to state PROBE by sending a unicast Neighbor Solicitation to TN1. The NUT did not update the state of TN1's NCE and stayed in state PROBE. The NUT retransmitted its unicast Neighbor Solicitation (Packet C) to TN1.
B. The NUT updated the NCE of TN1 to state STALE. After receiving the Echo Request from TN1, the NUT sent an Echo Reply and entered state DELAY. After DELAY_FIRST_PROBE_TIME, the NUT transitioned to state PROBE by sending a unicast Neighbor Solicitation to TN1. The NUT did not update the state of TN1's NCE and stayed in state PROBE. The NUT updated TN1's Link-layer address to its new link-layer address from the received Neighbor Solicitation (Packet C) and then updated the state of TN1's NCE to STALE. The NUT replied to the Neighbor Solicitation by sending a Neighbor Advertisement using TN1's new Link-Layer address and set the state of the TN1's Entry to DELAY. After DELAY_FIRST_PROBE_TIME, the NUT sent a unicast Neighbor Solicitation to TN1 using the new Link-layer address as the Target.
C. The NUT updated the NCE of TN1 to state STALE. After receiving the Echo Request from TN1, the NUT sent an Echo Reply and entered state DELAY. After DELAY_FIRST_PROBE_TIME, the NUT transitioned to state PROBE by sending a unicast Neighbor Solicitation to TN1. The NUT did not update the state of TN1's NCE and stayed in state PROBE. The NUT retransmitted its unicast Neighbor Solicitation (Packet D) to TN1.
D. The NUT updated the NCE of TN1 to state STALE. After receiving the Echo Request from TN1, the NUT sent an Echo Reply and entered state DELAY. After DELAY_FIRST_PROBE_TIME, the NUT transitioned to state PROBE by sending a unicast Neighbor Solicitation to TN1. The NUT did not update the state of TN1's NCE and stayed in state PROBE. The NUT updated TN1's Link-layer address to its new link-layer address from the received Neighbor Solicitation (Packet D) and then updated the state of TN1's NCE to STALE. The NUT replied to the Neighbor Solicitation by sending a Neighbor Advertisement using TN1's new Link-Layer

address and set the state of the TN1's Entry to DELAY. After DELAY_FIRST_PROBE_TIME, the NUT sent a unicast Neighbor Solicitation to TN1 using the new Link-layer address as the Target.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.14 | **Neighbor Solicitation Processing, Anycast (Routers Only)** | **A** | **PASS** |

**Purpose**: Verify that a router properly processes a Neighbor Solicitation for an anycast address.

**Comments on Test Procedure**

A. TN1 transmits a Neighbor Solicitation to the RUT's Subnet-Router anycast address.

**Comments on Test Results**

A. The RUT should responds to TN1 by sending a Neighbor Advertisement between 0 and MAX_ANYCAST_DELAY_TIME after it receives the Neighbor Solicitation. The RUT's Neighbor Advertisement should contain a value of 0 in the override flag field.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.15 | **Invalid Neighbor Advertisement Handling** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |
| | | **G** | **PASS** |

**Purpose**: Verify that a node takes the proper actions upon receipt of an invalid Neighbor Advertisement.

**Comments on Test Procedure**

A. *NUT receives invalid NA (Solicited Flag ==1):* TN1 transmits an Echo Request to the NUT. TN1 transmits Neighbor Advertisement (Packet A) with the Solicited flag set to 1.
B. *NUT receives invalid NA (Hop Limit == 254):* TN1 transmits an Echo Request to the NUT. TN1 transmits Neighbor Advertisement (Packet A) with the Hop Limit set to 254.
C. *NUT receives invalid NA (Invalid Checksum):* TN1 transmits an Echo Request to the NUT. TN1 transmits Neighbor Advertisement (Packet A) with an invalid checksum.
D. *NUT receives invalid NA (ICMP code! = zero):* TN1 transmits an Echo Request to the NUT. TN1 transmits Neighbor Advertisement (Packet A) with the ICMP code set to 1.
E. *NUT receives invalid NA (ICMP length < 24 octets):* TN1 transmits an Echo Request to the NUT. TN1 transmits Neighbor Advertisement (Packet A) with the ICMP length set to 16.
F. *NUT receives invalid NA (target == multicast address):* TN1 transmits an Echo Request to the NUT. TN1 transmits Neighbor Advertisement (Packet A) with the Target Address set to the solicited multicast of TN1's Link-Local address.
G. *NUT receives invalid NA (option length ==zero):* TN1 transmits an Echo Request to the NUT. TN1 transmits Neighbor Advertisement (Packet A) with the Option length set to 0.

**Comments on Test Results**

A. The NUT transmitted a Neighbor Solicitation to TN1's solicited-node multicast address. The NUT then transmitted a Neighbor Solicitation to TN1's solicited-node multicast address.

B. The NUT transmitted a Neighbor Solicitation to TN1's solicited-node multicast address. The NUT then ignored the Neighbor Advertisement sent by TN1 and continued to transmit Neighbor Solicitations to TN1's solicited-node multicast address.
C. The NUT transmitted a Neighbor Solicitation to TN1's solicited-node multicast address. The NUT then ignored the Neighbor Advertisement sent by TN1 and continued to transmit Neighbor Solicitations to TN1's solicited-node multicast address.
D. The NUT transmitted a Neighbor Solicitation to TN1's solicited-node multicast address. The NUT then ignored the Neighbor Advertisement sent by TN1 and continued to transmit Neighbor Solicitations to TN1's solicited-node multicast address.
E. The NUT transmitted a Neighbor Solicitation to TN1's solicited-node multicast address. The NUT then ignored the Neighbor Advertisement sent by TN1 and continued to transmit Neighbor Solicitations to TN1's solicited-node multicast address.
F. The NUT transmitted a Neighbor Solicitation to TN1's solicited-node multicast address. The NUT then ignored the Neighbor Advertisement sent by TN1 and continued to transmit Neighbor Solicitations to TN1's solicited-node multicast address.
G. The NUT transmitted a Neighbor Solicitation to TN1's solicited-node multicast address. The NUT then ignored the Neighbor Advertisement sent by TN1 and continued to transmit Neighbor Solicitations to TN1's solicited-node multicast address.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.16 | **Neighbor Advertisement Processing, No NCE** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |
| | | **G** | **PASS** |
| | | **H** | **PASS** |

**Purpose**: Verify that a node silently discards a Neighbor Advertisement if the target does not have a Neighbor Cache entry.

**Comments on Test Procedure**

A. *Receiving NA with S = 0, O = 0, and TLLA:* TR1 transmits Neighbor Advertisement (Packet A), which after TN1 transmits an Echo Request to the NUT.
B. *Receiving NA with S = 0, O = 1, and TLLA:* TR1 transmits Neighbor Advertisement (Packet B), which after TN1 transmits an Echo Request to the NUT.
C. *Receiving NA with S = 1, O = 0, and TLLA:* TR1 transmits Neighbor Advertisement (Packet C), which after TN1 transmits an Echo Request to the NUT.
D. *Receiving NA with S = 1, O = 1, and TLLA:* TR1 transmits Neighbor Advertisement (Packet D), which after TN1 transmits an Echo Request to the NUT.
E. *Receiving NA with S = 0, O = 0, and NO TLLA:* TR1 transmits Neighbor Advertisement (Packet A) without the Target Link-layer Address Option, which after TN1 transmits an Echo Request to the NUT.
F. *Receiving NA with S = 0, O = 1, and NO TLLA:* TR1 transmits Neighbor Advertisement (Packet B) without the Target Link-layer Address Option, which after TN1 transmits an Echo Request to the NUT.
G. *Receiving NA with S = 1, O = 0, and NO TLLA:* TR1 transmits Neighbor Advertisement (Packet C) without the Target Link-layer Address Option, which after TN1 transmits an Echo Request to the NUT.
H. *Receiving NA with S = 1, O = 1, and NO TLLA:* TR1 transmits Neighbor Advertisement (Packet D) without the Target Link-layer Address Option, which after TN1 transmits an Echo Request to the NUT.



**Comments on Test Results**

A-H. For each part, after receiving the Neighbor Advertisement from TN1, the NUT did not transmit any packets and no NCE's be created for TR1. After receiving the Echo Request from TN1, the NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to INCOMPLETE. The NUT sent a multicast Neighbor Solicitation to TN1.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.17 | **Neighbor Advertisement Processing, NCE State INCOMPLETE** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |

**Purpose**: Verify that a node properly updates its Neighbor Cache from the INCOMPLETE state upon receipt of a Neighbor Advertisement.

**Comments on Test Procedure**

A. *Receiving NA with S = 1 and O = 1:* TN1 transmits an Echo Request to the NUT's Link-Local address. Following the Echo Request, after observing traffic transmitted by the NUT, TN1 transmits a Neighbor Advertisement with both the Solicited and Override flags set. Observe both the packets transmitted by the NUT and the NCE of TN1 on the NUT. TN1 then transmits an Echo Request. Lastly, again observe both the packets transmitted by the NUT and the NCE of TN1 on the NUT.

B. *Receiving NA with S = 1 and O = 0:* TN1 transmits an Echo Request to the NUT's Link-Local address. Following the Echo Request, after observing traffic transmitted by the NUT, TN1 transmits a Neighbor Advertisement with the Solicited flag set, and the Override flag clear. Observe both the packets transmitted by the NUT and the NCE of TN1 on the NUT. TN1 then transmits an Echo Request. Lastly, again observe both the packets transmitted by the NUT and the NCE of TN1 on the NUT.

C. *Receiving NA with S = 0 and O = 1:* TN1 transmits an Echo Request to the NUT's Link-Local address. Following the Echo Request, after observing traffic transmitted by the NUT, TN1 transmits a Neighbor Advertisement with the Solicited flag clear, and the Override flag set. Observe both the packets transmitted by the NUT and the NCE of TN1 on the NUT. TN1 then transmits an Echo Request. Lastly, again observe both the packets transmitted by the NUT and the NCE of TN1 on the NUT.

D. *Receiving NA with S = 0 and O = 0:* TN1 transmits an Echo Request to the NUT's Link-Local address. Following the Echo Request, after observing traffic transmitted by the NUT, TN1 transmits a Neighbor Advertisement with both the Solicited and Override flags clear. Observe both the packets transmitted by the NUT and the NCE of TN1 on the NUT. TN1 then transmits an Echo Request. Lastly, again observe both the packets transmitted by the NUT and the NCE of TN1 on the NUT.

E. *Receiving NA without Target Link-Layer Address Option:* TN1 transmits an Echo Request to the NUT's Link-Local address. Following the Echo Request, after observing traffic transmitted by the NUT, TN1 transmits a Neighbor Advertisement without any Target Link-Layer Address Option. Lastly, again observe both the packets transmitted by the NUT and the NCE of TN1 on the NUT.



**Comments on Test Results**

A. After it received the Echo Request from TN1, the NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to **INCOMPLETE**. The NUT sent a multicast Neighbor Solicitation to TN1. After it received the Neighbor Advertisement from TN1, the NUT sent the queued Echo Reply to TN1 and updated its NCE for TN1 with the received Target Link-layer Address and changed the state of the NCE to **REACHABLE**. The NUT sent only an Echo Reply to TN1 since the NUT is in state **REACHABLE**. After DELAY_FIRST_PROBE_TIME, the NUT did not send a unicast Neighbor Solicitation to TN1.

B. After it received the Echo Request from TN1, the NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to **INCOMPLETE**. The NUT sent a multicast Neighbor Solicitation to TN1. After it received the Neighbor Advertisement from TN1, the NUT sent the queued Echo Reply to TN1 and updated its NCE for TN1 with the received Target Link-layer Address and changed the state of the NCE to **REACHABLE**. The NUT sent only an Echo Reply to TN1 since the NUT is in state **REACHABLE**. After DELAY_FIRST_PROBE_TIME, the NUT did not send a unicast Neighbor Solicitation to TN1.

C. After it received the Echo Request from TN1, the NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to **INCOMPLETE**. The NUT sent a multicast Neighbor Solicitation to TN1. After it received the Neighbor Advertisement from TN1, the NUT sent the queued Echo Reply to TN1 and updated its

NCE for TN1 with the received Target Link-layer Address and changed the state of the NCE to **STALE**. The NUT sent only an Echo Reply to TN1 since the NUT is in state **STALE**. After DELAY_FIRST_PROBE_TIME, the NUT sent a unicast Neighbor Solicitation to TN1.

D. After it received the Echo Request from TN1, the NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to **INCOMPLETE**. The NUT sent a multicast Neighbor Solicitation to TN1. After it received the Neighbor Advertisement from TN1, the NUT sent the queued Echo Reply to TN1 and updated its NCE for TN1 with the received Target Link-layer Address and changed the state of the NCE to **STALE**. The NUT sent only an Echo Reply to TN1 since the NUT is in state **STALE**. After DELAY_FIRST_PROBE_TIME, the NUT sent a unicast Neighbor Solicitation to TN1.

E. After it received the Echo Request from TN1, the NUT created a Neighbor Cache Entry for TN1 and set the state of the Entry to **INCOMPLETE**. The NUT sent a multicast Neighbor Solicitation to TN1. The NUT ignored the NA transmitted by TN1. There was no change in the neighbor cache for TN1 as it should stay in state **INCOMPLETE**. The NUT continued to send multicast Neighbor Solicitations to TN1.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.18 | **Neighbor Advertisement Processing, NCE State REACHABLE** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |
| | | **G** | **PASS** |
| | | **H** | **PASS** |
| | | **I** | **FAIL** |
| | | **J** | **PASS** |
| | | **K** | **FAIL** |
| | | **L** | **PASS** |
| | | **M** | **PASS** |
| | | **N** | **PASS** |
| | | **O** | **FAIL** |
| | | **P** | **PASS** |

**Purpose**: Verify that a node properly updates its Neighbor Cache from the **REACHABLE** state upon receipt of a Neighbor Advertisement.

**Comments on Test Procedure**

| Destination | Solicited flag | Override flag | TLLA | New State | Update Link-Layer Address | Part |
|---|---|---|---|---|---|---|
| Unicast | clear | clear | none | REACHABLE | no | A |
| Unicast | clear | set | none | REACHABLE | no | B |
| Unicast | set | clear | none | REACHABLE | no | C |
| Unicast | set | set | none | REACHABLE | no | D |
| Unicast | clear | clear | same | REACHABLE | no | E |
| Unicast | clear | set | same | REACHABLE | no | F |
| Unicast | set | clear | same | REACHABLE | no | G |
| Unicast | set | set | same | REACHABLE | no | H |
| Unicast | clear | clear | different | STALE | no | I |
| Unicast | clear | set | different | STALE | yes | J |
| Unicast | set | clear | different | STALE | no | K |
| Unicast | set | set | different | REACHABLE | yes | L |
| Multicast | clear | clear | same | REACHABLE | no | M |
| Multicast | clear | set | same | REACHABLE | no | N |
| Multicast | clear | clear | different | STALE | no | O |
| Multicast | clear | set | different | STALE | yes | P |

A. TN1 transmits an Echo Request to the NUT's Link-Local address. Observe the packets transmitted by the NUT and the NCE of TN1. TN1 then transmits a solicited Neighbor Advertisement to the NUT. Again, observe the packets transmitted by the NUT and the NCE of TN1. TN1 then transmits a Neighbor Advertisement with fields set according to Part A in the above table. TN1 transmits an Echo Request. Check The NCE of TN! On the NUT and observe the packets transmitted by the NUT. Perform the common cleanup procedure, and repeat this procedure 15 times for Parts B through P, according to the table above.

**Comments on Test Results**

A. The NUT created a Neighbor Cache Entry for TN1 and set the state of the entry to **INCOMPLETE**, after it received the first Echo Request. This caused the NUT to send a multicast Neighbor Solicitation to TN1. After the first Neighbor Advertisement was received, the NUT went to state **REACHABLE**. Therefore, it replied and, after DELAY_FIRST_PROBE_TIME, the NUT did not send a unicast Neighbor Solicitation to TN1. After the NUT received the test Neighbor Advertisement according to the table above, and an Echo Request from TN1, it updated the state of TN1's NCE and the LLA according to the table. After receiving the second Echo Request, the NUT reacted as follows:

**Parts A-H and L-N to REACHABLE**

After receiving the Echo Request from TN1, the NUT should send an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT should not send a unicast Neighbor Solicitation to TN1.

**Part L**

The NUT's Echo Reply should be sent to the new updated link-layer destination address of TN1.

**Parts J and P to STALE**

The NUT's Echo Reply should be sent to the new updated link-layer destination address of TN1. The Neighbor Solicitation should use the new link-layer address in the Target field.

I,K,O. After receiving the Echo Request, the NUT sent an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT did not transmit a unicast Neighbor Solicitation to TN1, indicating that the NUT did not enter state STALE.

According to RFC 2461, Section 7.2.5: When a Neighbor Cache Entry is in state REACHABLE, the entry should be changed to state STALE upon receiving a Neighbor Advertisement based upon the Solicited and Override flags and whether or not the new Link Local Address is equal to the previously cached value.

Therefore, while the NUT was in state REACHABLE after receiving the Neighbor Advertisement with certain bits set, the Neighbor Cache Entry should have been updated to state STALE.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.19 | **Neighbor Advertisement Processing, NCE State STALE** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |
| | | **G** | **PASS** |
| | | **H** | **PASS** |
| | | **I** | **PASS** |
| | | **J** | **PASS** |
| | | **K** | **PASS** |
| | | **L** | **PASS** |
| | | **M** | **PASS** |
| | | **N** | **PASS** |
| | | **O** | **PASS** |
| | | **P** | **PASS** |

**Purpose**: Verify that a node properly updates its Neighbor Cache from the STALE state upon receipt of a Neighbor Advertisement.

**Comments on Test Procedure**

| Destination | Solicited flag | Override flag | TLLA | New State | Update Link-Layer Address | Part |
|---|---|---|---|---|---|---|
| Unicast | clear | clear | none | STALE | no | A |
| Unicast | clear | set | none | STALE | no | B |
| Unicast | set | clear | none | REACHABLE | no | C |
| Unicast | set | set | none | REACHABLE | no | D |
| Unicast | clear | clear | same | STALE | no | E |
| Unicast | clear | set | same | STALE | no | F |
| Unicast | set | clear | same | REACHABLE | no | G |
| Unicast | set | set | same | REACHABLE | no | H |
| Unicast | clear | clear | different | STALE | no | I |
| Unicast | clear | set | different | STALE | yes | J |
| Unicast | set | clear | different | STALE | no | K |
| Unicast | set | set | different | REACHABLE | yes | L |
| Multicast | clear | clear | same | STALE | no | M |
| Multicast | clear | set | same | STALE | no | N |
| Multicast | clear | clear | different | STALE | no | O |
| Multicast | clear | set | different | STALE | yes | P |

A. TN1 transmits an Echo Request to the NUT's Link-Local address. Observe the packets transmitted by the NUT and the NCE of TN1. TN1 then transmits a solicited Neighbor Advertisement to the NUT. Again, observe the packets transmitted by the NUT and the NCE of TN1. Wait (REACHABLE_TIME * MAX_RANDOM_FACTOR) seconds, and again check the NCE of TN1 on the NUT. TN1 then transmits a Neighbor Advertisement with fields set according to Part A in the above table. TN1 transmits an Echo Request. Check The NCE of TN1 on the NUT and observe the packets transmitted by the NUT. Perform the common cleanup procedure, and repeat this procedure 15 times for Parts B through P, according to the table above.

**Comments on Test Results**

A. The NUT created a Neighbor Cache Entry for TN1 and set the state of the entry to **INCOMPLETE**, after it received the first Echo Request. This caused the NUT to send a multicast Neighbor Solicitation to TN1. After the first Neighbor Advertisement was received, the NUT went to state **REACHABLE**. Therefore, it replied and, after DELAY_FIRST_PROBE_TIME, the NUT did not send a unicast Neighbor Solicitation to TN1. After waiting (REACHABLE_TIME * MAX_RANDOM_FACTOR) seconds the NUT transitioned to state **STALE**. After the NUT received the test Neighbor Advertisement according to the table above, and an Echo Request from TN1, it updated the state of TN1's NCE and the LLA according to the table. After receiving the second Echo Request, the NUT reacted as follows:

**Parts C,D,G,H and L to REACHABLE**

After receiving the Echo Request from TN1, the NUT should send an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT should not send a unicast Neighbor Solicitation to TN1.

**Part L**

The NUT's Echo Reply should be sent to the new updated link-layer destination address of TN1.

**Parts A,B,E,F,I-K, and M-P to STALE**

After receiving the Echo Request from TN1, the NUT should send an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT should send a unicast Neighbor Solicitation to TN1.

**Parts J and P**

The NUT's Echo Reply should be sent to the new updated link-layer destination address of TN1. The Neighbor Solicitation should use the new link-layer address in the Target field.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.20 | **Neighbor Advertisement Processing, NCE State PROBE** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |
| | | **G** | **PASS** |
| | | **H** | **PASS** |
| | | **I** | **PASS** |
| | | **J** | **PASS** |
| | | **K** | **PASS** |
| | | **L** | **PASS** |
| | | **M** | **PASS** |
| | | **N** | **PASS** |
| | | **O** | **PASS** |
| | | **P** | **PASS** |

**Purpose**: Verify that a node properly updates its Neighbor Cache from the PROBE state upon receipt of a Neighbor Advertisement.

**Comments on Test Procedure**

| Destination | Solicited flag | Override flag | TLLA | New State | Update Link-Layer Address | Part |
|---|---|---|---|---|---|---|
| Unicast | clear | clear | none | PROBE | no | A |
| Unicast | clear | set | none | PROBE | no | B |
| Unicast | set | clear | none | REACHABLE | no | C |
| Unicast | set | set | none | REACHABLE | no | D |
| Unicast | clear | clear | same | PROBE | no | E |
| Unicast | clear | set | same | PROBE | no | F |
| Unicast | set | clear | same | REACHABLE | no | G |
| Unicast | set | set | same | REACHABLE | no | H |
| Unicast | clear | clear | different | PROBE | no | I |
| Unicast | clear | set | different | STALE | yes | J |
| Unicast | set | clear | different | PROBE | no | K |
| Unicast | set | set | different | REACHABLE | yes | L |
| Multicast | clear | clear | same | PROBE | no | M |
| Multicast | clear | set | same | PROBE | no | N |
| Multicast | clear | clear | different | PROBE | no | O |
| Multicast | clear | set | different | STALE | yes | P |

A. TN1 transmits an Neighbor Advertisement to the NUT's Link-Local address, only the Override flag is set, and the target address is TN1's Link-Local address. TN1 then transmits a link-local Echo Request to the NUT. Check the NCE of the NUT and observe the packets being transmitted by the NUT. After waiting (DELAY_FIRST_PROBE_TIME) seconds, recheck the NCE and again observe the packets being transmitted NUT. TN1 then transmits a Neighbor Advertisement with fields set according to Part A in the above table. TN1 also transmits an Echo Request, however, only in parts C, D, G, H, J, L, P. Again check the NCE of TN1 on the NUT and also observe the packets being transmitted by the NUT. Perform the common cleanup procedure and repeat this procedure fifteen times for parts B through P.

**Comments on Test Results**

A. The NUT created a Neighbor Cache Entry for TN1 and set the state of the entry to **STALE**, after it received the first Neighbor Advertisement. It replied to the first Echo Request and transitioned to state **DELAY**. After waiting (DELAY_FIRST_PROBE_TIME) seconds the NUT transitioned to state **PROBE**, and transmitted a unicast Neighbor Solicitation to TN1. After the NUT received the test Neighbor Advertisement according to the table above, and possibly also an Echo Request from TN1, it updated the state of TN1's NCE and the LLA according to the table, and the NUT reacted as follows:

**Parts C,D,G,H and L to REACHABLE**

After receiving the Echo Request from TN1, the NUT should send an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT should not send a unicast Neighbor Solicitation to TN1.

**Part L**

The NUT's Echo Reply should be sent to the new updated link-layer destination address of TN1.

**Parts J and P to STALE**

After receiving the Echo Request from TN1, the NUT should send an Echo Reply. After DELAY_FIRST_PROBE_TIME, the NUT should send a unicast Neighbor Solicitation to TN1.

The NUT's Echo Reply should be sent to the new updated link-layer destination address of TN1. The Neighbor Solicitation should use the new link-layer address in the Target field.

**Parts A,B,E,F,I,K, and M-0 to PROBE**

The NUT should send a unicast Neighbor Solicitation to TN1.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.1.21 | **Neighbor Advertisement Processing, R-bit Change (Hosts Only)** | **A-D** | **N/A** |

**Purpose**: Verify that a host takes appropriate actions when a neighbor who is a router starts transmitting Neighbor Advertisements with the Router flag clear.

**Comments on Test Procedure**

*Setup for each part:* Configure a default route with TR1 as the next hop. TR1 then transmits an Echo Request to the RUT. Observe the packets sent by the RUT. TR1 transmits a Neighbor Advertisement. After an Echo Reply is received setup is complete.

A. TR1 transmits a Router Advertisement without a Source Link-layer Address Option and the following values: Router Lifetime = 20 seconds; Reachable Time = 100 seconds; Retransmit Interval = 1 second; Prefix = TR1's Global Prefix. TN1 then transmits an Echo Request destined for the HUT's Global Address. Observe the packets sent by the HUT. TR1 responds to any Neighbor Solicitation from the HUT with a Neighbor Advertisement with the Router, Solicited, and Override flags set. Observe the packets sent by the HUT. TR1 transmits a Neighbor Advertisement, which has only the Router Flag not set. TN1 again transmits an Echo Request. Observe the packets sent by the HUT. Perform the common cleanup procedure.

B. TR1 transmits a Router Advertisement without a Source Link-layer Address Option and the following values: Router Lifetime = 20 seconds; Reachable Time = 100 seconds; Retransmit Interval = 1 second; Prefix = TR1's Global Prefix. TN1 then transmits an Echo Request destined for the HUT's Global Address. Observe the packets sent by the HUT. TR1 responds to any Neighbor Solicitation from the HUT with a Neighbor Advertisement with the Router, Solicited, and Override flags set. Observe the packets sent by the HUT. TR1 transmits a Neighbor Advertisement, which has no flags set. TN1 again transmits an Echo Request. Observe the packets sent by the HUT. Perform the common cleanup procedure.

C. TR1 transmits a Router Advertisement without a Source Link-layer Address Option and the following values: Router Lifetime = 20 seconds; Reachable Time = 100 seconds; Retransmit Interval = 1 second; Prefix = TR1's Global Prefix. TN1 then transmits an Echo Request destined for the HUT's Global Address. Observe the packets sent by the HUT. TR1 responds to any Neighbor Solicitation from the HUT with a Neighbor Advertisement with the Router, Solicited, and Override flags set. Observe the packets sent by the HUT. TR1 transmits a Neighbor Advertisement, which has only the Override Flag set. TN1 again transmits an Echo Request. Observe the packets sent by the HUT. Perform the common cleanup procedure.

D. TR1 transmits a Router Advertisement without a Source Link-layer Address Option and the following values: Router Lifetime = 20 seconds; Reachable Time = 100 seconds; Retransmit Interval = 1 second; Prefix = TR1's Global Prefix. TN1 then transmits an Echo Request destined for the HUT's Global Address. Observe the packets sent by the HUT. TR1 responds to any Neighbor Solicitation from the HUT with a Neighbor Advertisement with the Router, Solicited, and Override flags set. Observe the packets sent by the HUT. TR1 transmits a Neighbor Advertisement, which has only the Solicited Flag set. TN1 again transmits an Echo Request. Observe the packets sent by the HUT. Perform the common cleanup procedure.



**Comments on Test Results**

These tests are performed on Hosts only.

## Group 2: Router and Prefix Discovery

The following tests cover Router and Prefix Discovery in IPv6.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.1 | **Router Solicitation (Hosts Only)** | **A** | **N/A** |
| **Purpose**: Verify that a host sends valid Router Solicitations at the appropriate time. | | | |
| **Comments on Test Procedure** | | | |
| A. Reboot the HUT. Observe the packets transmitted by the HUT. | | | |
| **Comments on Test Results** | | | |
| A. This test is performed on Hosts only. | | | |

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.2 | **Router Solicitations, Solicited Router Advertisement (Host Only)** | **A-F** | **N/A** |

**Purpose**: Verify that a host sends valid Router Solicitations at the appropriate time.

**Comments on Test Procedure**

A. *Valid Router Advertisement, No Source Link-layer Address Option*: Reboot the HUT. After the HUT transmits a Router Solicitation, TR1 transmits a Router Advertisement without a Source Link-layer Address Option. The Source Address is the Link-Local address of TR1. The Hop Limit is 255, the ICMP Code is 0, the ICMP Checksum is valid. After waiting RTR_SOLICITATION_INTERVAL+MAX_RTR_SOLICITATION_DELAY, observe the packets transmitted from the HUT.

B. *Valid Router Advertisement, Source Link-layer Address Option*: Reboot the HUT. After the HUT transmits a Router Solicitation, TR1 transmits a Router Advertisement with a Source Link-layer Address Option. The Source Address is the Link-Local address of TR1. The Hop Limit is 255, the ICMP Code is 0, the ICMP Checksum is valid. After waiting RTR_SOLICITATION_INTERVAL+MAX_RTR_SOLICITATION_DELAY, observe the packets transmitted from the HUT.

C. *Invalid Router Advertisement, Global Source Address*: Reboot the HUT. After the HUT transmits a Router Solicitation, TR1 transmits a Router Advertisement with a Source Address equal to the Global Address of TR1, but is valid elsewhere. The Hop Limit is 255, the ICMP Code is 0, the ICMP Checksum is valid. After waiting RTR_SOLICITATION_INTERVAL+MAX_RTR_SOLICITATION_DELAY, observe the packets transmitted from the HUT.

D. *Invalid Router Advertisement, Bad Hop Limit*: Reboot the HUT. After the HUT transmits a Router Solicitation, TR1 transmits a Router Advertisement without a Source Link-layer Address Option. The Source Address is the Link-Local address of TR1. The Hop Limit is 2, the ICMP Code is 0, the ICMP Checksum is valid. After waiting RTR_SOLICITATION_INTERVAL+MAX_RTR_SOLICITATION_DELAY, observe the packets transmitted from the HUT.

E. *Invalid Router Advertisement, Bad ICMP Checksum*: Reboot the HUT. After the HUT transmits a Router Solicitation, TR1 transmits a Router Advertisement without a Source Link-layer Address Option. The Source Address is the Link-Local address of TR1. The Hop Limit is 255, the ICMP Code is 0, the ICMP Checksum is invalid. After waiting RTR_SOLICITATION_INTERVAL+MAX_RTR_SOLICITATION_DELAY, observe the packets transmitted from the HUT.

F. *Invalid Router Advertisement, Bad ICMP Code*: Reboot the HUT. After the HUT transmits a Router Solicitation, TR1 transmits a Router Advertisement without a Source Link-layer Address Option. The Source Address is the Link-Local address of TR1. The Hop Limit is 255, the ICMP Code is 1, the ICMP Checksum is valid. After waiting RTR_SOLICITATION_INTERVAL+MAX_RTR_SOLICITATION_DELAY, observe the packets transmitted from the HUT.



**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.3 | **Host Ignores Router Solicitations (Host Only)** | **A-C** | **N/A** |

**Purpose**: Verify that a host ignores Router Solicitations and does not update its Neighbor Cache.

**Comments on Test Procedure**

A. *All-Router Multicast Destination*: TN1 transmits a Router Solicitation with a Source Link-layer Address Option to the All-Router multicast address. After waiting (REATRANS_TIMER * MAX_ * CAST_SOLICIT) (3 seconds), TN1 transmits a link-local Echo Request to the HUT. After waiting 2 seconds, observe the packets transmitted by the HUT.
B. *All-Nodes Multicast Destination*: TN1 transmits a Router Solicitation with a Source Link-layer Address Option to the All-Nodes multicast address. After waiting (REATRANS_TIMER * MAX_ * CAST_SOLICIT) (3 seconds), TN1 transmits a link-local Echo Request to the HUT. After waiting 2 seconds, observe the packets transmitted by the HUT.
C. *Link-Local Unicast Destination*: TN1 transmits a Router Solicitation with a Source Link-layer Address Option to the Link-Local address of the HUT. After waiting (REATRANS_TIMER * MAX_ * CAST_SOLICIT) (3 seconds), TN1 transmits a link-local Echo Request to the HUT. After waiting 2 seconds, observe the packets transmitted by the HUT.

**Comments on Test Results**

These tests are performed on Hosts only.



| | | | |
|---|---|---|---|
| v6LC.2.2.4 | **Router Ignores Invalid Router Solicitations (Routers Only)** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |

**Purpose**: Verify that a router ignores invalid Router Solicitations.

**Comments on Test Procedure**

A. *Hop Limit is not 255:* TN1 transmits a Router Solicitation with an IPv6 Hop Limit of 254. Otherwise, the Router Solicitation is valid. Observe the packets transmitted by the RUT.
B. *ICMPv6 checksum is not valid:* TN1 transmits a Router Solicitation with an invalid ICMPv6 checksum. Otherwise, the Router Solicitation is valid. Observe the packets transmitted by the RUT.
C. *ICMPv6 code is not 0:* TN1 transmits a Router Solicitation with an invalid ICMPv6 code of 1. Otherwise, the Router Solicitation is valid. Observe the packets transmitted by the RUT.
D. *ICMPv6 length is less than 8 Octets:* TN1 transmits a Router Solicitation with an ICMPv6 length of 6. Otherwise, the Router Solicitation is valid. Observe the packets transmitted by the RUT.
E. *Option has length 0:* TN1 transmits a Router Solicitation that contains an Option with length 0. Otherwise, the Router Solicitation is valid. Observe the packets transmitted by the RUT.
F. *Unspecified IP source address and a source link-layer address option:* TN1 transmits a Router Solicitation with an unspecified source address and a Source Link-layer Address Option. Otherwise, the Router Solicitation is valid. Observe the packets transmitted by the RUT.

**Comments on Test Results**

A-F. The RUT discarded the Router Solicitation from TN1 and did not transmit a corresponding Router Advertisement within MAX_RA_DELAY_TIME (0.5) seconds.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.5 | **Router Sends Valid Router Advertisement (Routers Only)** | **A** | **PASS** |

**Purpose**: Verify that a router sends valid Router Advertisements.

**Comments on Test Procedure**

A. TN1 transmits a valid Router Solicitation. Observe the packets transmitted by the RUT.

**Comments on Test Results**

A. The RUT transmitted valid Router Advertisements that satisfied all of the following validity checks:
- IP Source Address is a link-local address.
- The IP Hop Limit field has a value of 255, i.e., the packet could not possibly have been forwarded by a router.
- If the message includes an IP Authentication Header, the message authenticates correctly.
- ICMP Checksum is valid.
- ICMP Code is 0.
- ICMP length (derived from the IP length) is 16 or more octets.
- All included options have a length that is greater than zero.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.6 | **Router Does Not Send Router Advertisements on Non-advertising Interface (Routers Only)** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a router does not send Router Advertisements on non-advertising interfaces.

**Comments on Test Procedure**

A. *No advertising interface:* Configure one interface (Interface A) on the RUT to be a non-advertising interface. Configure TR1 to transmit a Router Solicitation to the RUT on Interface A. Observe the packets transmitted by the RUT on Interface A.
B. *Advertising interface*: If the RUT supports two network interface, configure the first (Interface A) to be an advertising interface, and the second (Interface B) to be a non-advertising interface.

**Comments on Test Results**

A. The RUT did not send any Router Advertisements out on Interface A (referenced above).
B. The RUT sent out Router Advertisements out on Interface A (referenced above), and did not send any Router Advertisements out on Interface B (referenced above).

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.7 | **Sending Unsolicited Router Advertisements (Routers Only)** | **A** | **PASS** |

**Purpose**: Verify that a router sends the first few advertisements (up to MAX_INITIAL_RTR_ADVERTISEMENTS) from an interface when it becomes an advertising interface at an maximum interval value of MAX_INITIAL_RTR_ADVERT_INTERVAL (16) seconds.

**Comments on Test Procedure**

A. Configure an interface on the RUT to be an advertising interface with a MinRtrAdvInterval of 5 seconds and a MaxRtrInterval of 10 seconds, and observe the packets transmitted out on this interface.

**Comments on Test Results**

A. The RUT transmitted the first MAX_INITIAL_RTR_ADVERTISEMENT (3) at randomly chosen intervals of no greater than MAX_INITIAL_RTR_ADVERT_INTERVAL (16) seconds. After that, the RUT transmitted Router Advertisements at randomly chosen intervals that lie between the interface's configure MinRtrAdvInterval (5) and MaxRtrAdvInterval (10) and it did not transmit Router Advertisements more frequently than indicated by MinRtrAdvInterval (5) seconds.



| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.8 | **Ceasing to Be An Advertising Interface (Routers Only)** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a router sends correct Router Advertisements when its interface ceases to be an advertising interface.

**Comments on Test Procedure**

A. *Ceasing to be an advertising interface:* Configure an interface on the RUT to be an advertising interface. After a few seconds, configure the same interface to discontinue being a adverting interface.
B. *Disabled IP forwarding capability:* Configure an interface on the RUT to be an advertising interface and disable the RUT's IP forwarding capability.

**Comments on Test Results**

A. The RUT SHOULD have, upon disabling Router Advertisements on the interface, sent out no more than MAX_FINAL_RTR_ADVERTISEMENTS with a Router Lifetime field of zero.

B. The RUT SHOULD have, upon disabling IP forwarding capabilities, transmitted all subsequent Router Advertisements with a Router Lifetime Field of zero.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.9 | **Processing Router Solicitations (Routers Only)** | **A** | **PASS** |
| | | **B** | **FAIL** |

**Purpose**: Verify that a router correctly processes Router Solicitations and transmits Router Advertisements.

**Comments on Test Procedure**

A. *MAX_RA_DELAY_TIME:* TN1 transmits Router Solicitation A twice, 3 seconds apart. The Destination Address is the all-routers multicast address.
B. *MIN_DELAY_BETWEEN_RAS:* Configure the RUT with a MinRtrAdvInterval of 30 seconds and a MaxRtrAdvInterval of 40 seconds. TN1 transmits Router Solicitation B twice, 2 seconds apart. The destination Address is the all-routers multicast address.

**Comments on Test Results**

A. The RUT transmitted a Router Advertisement between 0 and MAX_RA_DELAY_TIME (0.5) seconds after the receipt of each Router Solicitation A.
B. The RUT incorrectly transmitted more than one advertisement every MIN_DELAY_BETWEEN_RAS (3) seconds.

According to RFC 2461, Section 6.2.6: "Router Solicitations in which the Source Address is the unspecified address MUST NOT update the router's Neighbor Cache."

Therefore, the RUT should have continued to transmit advertisements at least MIN_DELAY_BETWEEN_RAS (3) seconds apart.



| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.10 | **Default Router Switch (Hosts Only)** | **A** | **N/A** |

**Purpose**: Verify that a host maintains at least two routers in its Default Router List and will switch routers when the router in use fails.

**Comments on Test Procedure**

A. TR1 transmits Router Advertisement A with a Retransmit Interval of 1 second and the L-Bit set to 1. A few seconds after, TR1 transmits Packet A, an Echo Request. TR1 transmits a Neighbor Advertisement in response to any Neighbor Solicitations from the HUT. TR2 transmits Router Advertisement B with Retransmit Interval of 1 second and the L-Bit set to 1. When Reachable Time expires, and the HUT solicits TR1, no Neighbor Advertisements are transmitted by TR1.

**Comments on Test Results**

A. This test is performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.11 | **Router Advertisement Processing, Validity (Hosts Only)** | **A-F** | **N/A** |

**Purpose**: Verify that a host properly discards an invalid Router Advertisement.

**Comments on Test Procedure**

A. *Global Source Address:* TR1 transmits the Router Advertisement. The Source Address is the global address of TR1. The Router Advertisements is valid otherwise. Then TR1 transmits a link-local Echo Request to the HUT.
B. *Hop Limit less than 255:* TR1 transmits the Router Advertisement. The Hop Limit is 2. The Router Advertisement is valid otherwise. Then TR1 transmits a link-local Echo Request to the HUT.
C. *Invalid Checksum:* TR1 transmits the Router Advertisement. The ICMP Checksum is invalid. The Router Advertisement is valid otherwise. Then TR1 transmits a link-local Echo Request to the HUT.
D. *Invalid ICMP Code:* TR1 transmits the Router Advertisement. The ICMP Code is 1. The Router Advertisement is valid otherwise. Then TR1 transmits a link-local Echo Request to the HUT.
E. *Invalid ICMP Length:* TR1 transmits the Router Advertisement with an ICMP length of 14. The Router Advertisement is valid otherwise. Then TR1 transmits a link-local Echo Request to the HUT.
F. *Option of Length 0:* TR1 transmits the Router Advertisement with an option of length 0. The Router Advertisement is valid otherwise. Then TR1 transmits a link-local Echo Request to the HUT.

**Comments on Test Results**

These tests are performed on Hosts only.



| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.12 | **Router Advertisement Processing, Cur Hop Limit** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a host properly processes the Cur Hop Limit field of a Router Advertisement.

**Comments on Test Procedure**

A. *Unspecified:* TN1 transmits an Echo Request to the NUT. TR1 transmits a Router Advertisement with a Cur Hop Limit value of 0 (Zero) followed by an Echo Request to the NUT.
B. *Non-Zero:* TN1 transmits an Echo Request to the NUT. TR1 transmits a Router Advertisement with a Cur Hop Limit value of 15 followed by an Echo Request to the NUT.

**Comments on Test Results**

A. The NUT responded to the Request from TN1. The NUT responded again to the Request from TN1. The Hop Limit value in the Echo Reply should be the same as was used in step 2.
B. The NUT should have replied to the second Echo Request from TN1 with a Hop Limit value of 15 in the Echo Reply.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.13 | **Router Advertisement Processing, Router Lifetime (Hosts Only)** | **A-C** | **N/A** |

**Purpose**: Verify that a host properly processes a Router Advertisement and the Router Lifetime field within it.

**Comments on Test Procedure**

A. *Router Lifetime Updated with Same Lifetime:* TR1 transmits the Router Advertisement with Router Lifetime of 20 seconds and Reachable time of 600 seconds. TN2 transmits a global Echo Request to the HUT every second for 19 seconds. TR1 transmits the same Router Advertisement as before. TN2 transmits a global Echo Request to the HUT every second for 21 seconds.
B. *Router Lifetime Set to Zero:* TN2 transmits a global Echo Request to the HUT. Shortly after, TR1 transmits a Router Advertisement with Router Lifetime set to zero followed by TN2 transmits a global Echo Request to the HUT. TR2 transmits a Router Advertisement with Router Lifetime set to zero and TN2 transmits a global Echo Request to the HUT. TR3 transmits a Router Advertisement with Router Lifetime set to zero and then TN2 transmits a global Echo Request to the HUT.
C. *Router Lifetime Set to Five; Allowed to Expire:* TN2 transmits a global Echo Request to the HUT. Shortly after, TR1 transmits a Router Advertisement with Router Lifetime set to 5 followed by TN2 transmits a global Echo Request to the HUT. TR2 transmits a Router Advertisement with Router Lifetime set to 5 and TN2 transmits a global Echo Request to the HUT. TR3 transmits a Router Advertisement with Router Lifetime set to 5 and then TN2 transmits a global Echo Request to the HUT.

**Comments on Test Results**

These tests are performed on Hosts only.



| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.14 | **Router Advertisement Processing, Reachable Time** | **A** | **N/A** |
| | | **B** | **PASS** |

**Purpose**: Verify that a host updates its BaseReachableTime variable and re-computes its ReachableTime variable upon receipt of a Router Advertisement with a specified Reachable Time.

**Comments on Test Procedure**

A. *RA Processing – Reachable Time (Host Only):* TR1 transmits the Router Advertisement with a Router Lifetime of 0 seconds and a Reachable Time of 10 seconds. TN1 transmits a link-local Echo Request to the HUT every second for 40 seconds. TN1 must reply to any Neighbor Solicitations from the HUT. TR1then transmits the Router Advertisement with a Reachable Time of 40 seconds. TN1 transmits a link-local Echo Request to the HUT every second for 140 seconds.
B. *Reachable Time Configuration (Routers Only):* Configure the RUT to transmit Router Advertisements with a Router Lifetime value of 0 seconds and a Reachable Time of 10 seconds. TN1 transmits a link-local Echo Request to the RUT. TN1 must reply to any Neighbor Solicitations from the RUT. TN1 transmits a link-local Echo Request to the HUT every second for 40 seconds.

**Comments on Test Results**

A. This test is performed on Hosts only.
B. The RUT should solicit for TN1's link-local address and transmit an Echo Reply. The RUT should transmit a Neighbor Solicitation with a Target Address of TN1's link-local address at an interval between 5 and 15 seconds.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.2.15 | **Router Advertisement Processing, Neighbor Cache (Hosts Only)** | **A-F** | **N/A** |

**Purpose**: Verify that a host properly updates its Neighbor Cache upon receipt of a Router Advertisement.

**Comments on Test Procedure**

A. *RA processing, no NCE:* TR1 transmits Router Advertisement A with Reachable Time set to 10 seconds and Retransmit Interval set to 1 second. TR1 transmits an Echo Request to the HUT.
B. *RA processing, NCE INCOMPLETE:* TR1 transmits Echo Request B. TR1 does not respond to any Neighbor Solicitations from the HUT. TR1 transmits Router Advertisement A with Reachable Time set to 10 seconds and Retransmit Interval set to 1 second.
C. *RA with SLLA changed, NCE REACHABLE:* TR1 transmits Echo Request B. TR1 does not respond to any Neighbor Solicitations from the HUT followed by TR1 transmitting Neighbor Advertisement C. TR1 transmits Router Advertisement A with a different Source Link-layer Address. TR1 transmits an Echo Request to the HUT.
D. *RA with SLLA unchanged, NCE REACHABLE:* TR1 transmits Echo Request B. TR1 does not respond to any Neighbor Solicitations from the HUT followed by TR1 transmitting Neighbor Advertisement C. TR1 transmits Router Advertisement A with the same Source Link-layer Address. TR1 transmits an Echo Request to the HUT.
E. *RA with SLLA changed, NCE PROBE: TR1* transmits Echo Request B. TR1 does not respond to any Neighbor Solicitations from the HUT followed by TR1 transmitting Neighbor Advertisement C. TR1 then transmits Echo Request B. TR1 transmits Router Advertisement A with a different Source Link-layer Address. TR1 transmits an Echo Request to the HUT.
F. *RA with SLLA unchanged, NCE PROBE: TR1* transmits Echo Request B. TR1 does not respond to any Neighbor Solicitations from the HUT followed by TR1 transmitting Neighbor Advertisement C. TR1 then transmits Echo Request B. TR1 transmits Router Advertisement A with the same Source Link-layer Address. TR1 transmits an Echo Request to the HUT.

**Comments on Test Results**

These test are performed on Hosts only.

## Group 3: Redirect Function

The following tests cover the Redirect function in IPv6.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.1 | **Redirected On-link: Valid (Hosts Only)** | **A-H** | **N/A** |
| **Purpose**: Verify that a host properly processes valid Redirect messages when redirected on-link. | | | |
| **Comments on Test Procedure** | | | |

| IPv6 Destination Address | TLLA Option | Redirected Packet Option | Part |
|---|---|---|---|
| Link-local (NUT) | No | No | A |
| Link-local (NUT) | No | Yes | B |
| Link-local (NUT) | Yes | No | C |
| Link-local (NUT) | Yes | Yes | D |
| Global (NUT) | No | No | E |
| Global (NUT) | No | Yes | F |
| Global (NUT) | Yes | No | G |
| Global (NUT) | Yes | Yes | H |

Part A. through H.

*Destination Addresses, TLLA Options, and Redirected Packet Options:* TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The DestinationAddress is the global address of the HUT. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TN1. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above. TN1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.



**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.2 | **Redirected On-link: Suspicious (Hosts Only)** | **A-C** | **N/A** |

**Purpose**: Verify that a host properly processes suspicious Redirect messages when redirected on-link.

**Comments on Test Procedure**

A. *Option Unrecognized:* TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the global address of TN1. The Redirect message contains a Target Link-layer Address option. The Redirect message also contains an unrecognized option. TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

B. *Reserved Field is Non-zero:* TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the global address of TN1. The Redirect message contains a Target Link-layer Address option and has a non-zero Reserved field. The Redirect message also contains an unrecognized option. TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

C. *Target Address not Covered by On-link Prefix:* TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the global address of TN1. The Redirect message contains a Target Link-layer Address option but the Target Address of the global address of TN1 is not covered by an on-link prefix.. The Redirect message also contains an unrecognized option. TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.3 | **Redirected On-link: Invalid (Hosts Only)** | **A-I** | **N/A** |

**Purpose**: Verify that a host properly processes invalid Redirect messages when redirected on-link.

**Comments on Test Procedure**

A. *Redirect Source Address is Global:* TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR1 then transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TN1. The Redirect message contains an incorrect IPv6 Source Address (the off-link global address of TN2). Next, TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

B. *Redirect Source Address is not the current first-hop router:* TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR1 then transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TN1. The Redirect message contains an incorrect IPv6 Source Address (the link-local address of TR2). Next, TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

C. *Hop Limit is not 255:* TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR1 then transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TN1. The Redirect message contains an incorrect IPv6 Hop Limit of 254. Next, TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

D. *ICMPv6 Code is not 0:* TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR1 then transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TN1. The Redirect message contains an incorrect ICMPv6 Code of 1. Next, TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

E. *ICMPv6 Checksum is invalid:* TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR1 then transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TN1. The Redirect message contains an incorrect ICMPv6 Checksum. Next, TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

F. *ICMPv6 Destination Address is Multicast:* TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR1 then transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TN1. The Redirect message contains an ICMPv6 Destination Address of the All-nodes multicast address. Next, TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

G. *Target Address is Multicast:* TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR1 then transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TN1. The Redirect message contains an ICMPv6 Target Address of the All-nodes multicast address. Next, TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

H. *ICMPv6 length is less than 40 Octets:* TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR1 then transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TN1. The Redirect message contains an invalid ICMPv6 Length of 39 bytes. Next, TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

I. *Option has Length Zero:* TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR1 then transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TN1. The Redirect message contains an Option with length 0. Next, TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.4 | **Redirected to Alternate Router: Valid (Hosts Only)** | **A-I** | **N/A** |

**Purpose**: Verify that a host properly processes valid Redirect messages when redirected on-link.

**Comments on Test Procedure**

| IPv6 Destination Address | TLLA Option | Redirected Packet Option | Part |
|---|---|---|---|
| Link-local (NUT) | No | No | A |
| Link-local (NUT) | No | Yes | B |
| Link-local (NUT) | Yes | No | C |
| Link-local (NUT) | Yes | Yes | D |
| Global (NUT) | No | No | E |
| Global (NUT) | No | Yes | F |
| Global (NUT) | Yes | No | G |
| Global (NUT) | Yes | Yes | H |

Parts A-H: Destination *Addresses, TLLA Options, and Redirected Packet Options:* TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. Next, TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. Following is, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above. Shortly After, TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

I: *Redirected to Router not in Default Router List:* TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. Next, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above. Shortly after, TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.5 | **Redirected to Alternate Router: Suspicious (Hosts only)** | **A-C** | **N/A** |

**Purpose**: Verify that a host properly processes suspicious Redirect messages when redirected on-link.

**Comments on Test Procedure**

A. *Option Unrecognized*: TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 then forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. After, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option. The Redirect message also contains an unrecognized option. TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

B. *Reserved Field is Non-zero:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 then forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. After, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option. The Redirect message also contains a non-zero Reserved field. TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

C. *Target Address not Covered by On-link Prefix:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 then forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. After, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option. The Redirect message also contains a Target Address of a global address of TR2 that is not covered by an on-link prefix. TR1 forwards an Echo Request to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.



**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.6 | **Redirected to Alternate Router: Invalid (Hosts Only)** | **A-I** | **N/A** |
| **Purpose**: Verify that a host properly processes invalid Redirect messages when redirected on-link. | | | |
| **Comments on Test Procedure** | | | |

A. *Redirect Source Address is Global*: TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. After, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an incorrect IPv6 Source Address (the off-link global address of TN2). TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

B. *Redirect Source Address is not the current first-hop router:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. After, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an incorrect IPv6 Source Address (the off-link global address of TR2). TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

C. *Hop Limit is not 255:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. After, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an incorrect IPv6 Hop Limit of 254. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

D. *ICMPv6 Code is not 0:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. After, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an incorrect ICMPv6 Code of 1. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

E. *ICMPv6 Checksum is invalid:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. After, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an incorrect ICMPv6 Checksum. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

F. *ICMPv6 Destination Address is Multicast:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. After, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an ICMPv6 Destination Address of the All-nodes multicast address. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

G. *Target Address is Multicast:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. After, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains a Target Address of the All-nodes multicast address. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

H. *ICMPv6 length is less than 40 Octets:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. After, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an invalid IPv6 Length of 39 bytes. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

I. *Option has Length Zero: TR1* transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. After, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an Option with length 0. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.7 | **Redirected Twice (Hosts Only)** | **A** | **N/A** |

**Purpose**: Verify that a host properly processes valid Redirect messages twice for the same destination.

**Comments on Test Procedure**

A. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. TR1 also forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR3 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. TR2 also transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR3. After, TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

**Comments on Test Results**

A. This tests is performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.8 | **Invalid Option (Hosts Only)** | **A-C** | **N/A** |

**Purpose**: Verify that a host ignores invalid options in Redirect messages and processes the remainder of the Redirect normally.

**Comments on Test Procedure**

A. *Path MTU Option:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. TR1also transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains a Path MTU option. Afterwards, TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

B. *Prefix Information Option:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. TR1also transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains a Prefix Information option. Afterwards, TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.

C. *Source Link-layer Address Option:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT. TR2 transmits a Router Advertisement with a non-zero Router Lifetime and a Source Link-layer Address option. TR1also transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains a Source Link-layer Address option. Afterwards, TR1 forwards an Echo Request from TN1 to the HUT. The Source Address is the off-link global address of TN1. The Destination Address is the global address of the HUT.



**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.9 | **No Destination Cache Entry (Hosts Only)** | **A** | **N/A** |

**Purpose**: Verify that a host properly processes a Redirect message when there is no entry for the destination in the host's Destination Cache.

**Comments on Test Procedure**

A. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-Layer option with the link-layer address of TR2. Afterwards, TR1 forwards an Echo Request from TN1 to the HUT. The IPv6 Source Address is the off-link global address of TN1. The IPv6 Destination Address is the global address of the HUT.

**Comments on Test Results**

A. This tests is performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.10 | **Neighbor Cache Updated, No Neighbor Cache Entry (Hosts Only)** | **A-D** | **N/A** |

**Purpose**: Verify that a host properly updates its Neighbor Cache entry upon receipt of a valid ICMP Redirect Message.

**Comments on Test Procedure**

| TLLA Option | Redirected Packet Option | New NC State | Link-layer Address | Part |
|---|---|---|---|---|
| No | No | No NCE | Unchanged | A |
| Yes | No | STALE | Updated | B |
| Yes | Yes | STALE | Updated | C |
| Yes | Yes, packet > 1280 | STALE | Updated | D |

A. *No TLLA Option, No Redirected Packet Option, Link-layer Address Unchanged:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above.

B. *TLLA Option, No Redirected Packet Option, Link-layer Address Updated:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above.

C. *TLLA Option, Redirected Packet Option, Link-layer Address Updated:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above.

D. *TLLA Option, Oversized Redirected Packet Option, Link-layer Address Updated:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above.



**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.11 | **Neighbor Cache Updated from State INCOMPLETE (Hosts Only)** | **A-D** | **N/A** |

**Purpose**: Verify that a host properly updates its Neighbor Cache entry upon receipt of a valid ICMP Redirect Message.

**Comments on Test Procedure**

| TLLA Option | Redirected Packet Option | New NC State | Link-layer Address | Part |
|---|---|---|---|---|
| No | No | INCOMPLETE | Unchanged | A |
| Yes | No | STALE | Updated | B |
| Yes | Yes | STALE | Updated | C |
| Yes | Yes, packet > 1280 | STALE | Updated | D |

A. *No TLLA Option, No Redirected Packet Option, Link-layer Address Unchanged*: TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR2 transmits a link-local Echo Request to the HUT. TR2 does not reply to Neighbor Solicitations. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above.

B. *TLLA Option, No Redirected Packet Option, Link-layer Address Updated*: TR2 transmits a link-local Echo Request to the HUT. TR2 does not reply to Neighbor Solicitations. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above.

C. *TLLA Option, Redirected Packet Option, Link-layer Address Updated:* TR2 transmits a link-local Echo Request to the HUT. TR2 does not reply to Neighbor Solicitations. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above.

D. *TLLA Option, Oversized Redirected Packet Option, Link-layer Address Updated:* TR2 transmits a link-local Echo Request to the HUT. TR2 does not reply to Neighbor Solicitations. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above.



**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.12 | **Neighbor Cache Updated from State REACHABLE (Hosts Only)** | **A-E** | **N/A** |

**Purpose**: Verify that a host properly processes a Redirect message when there is no entry for the destination in the host's Destination Cache.

**Comments on Test Procedure**

| TLLA Option | Redirected Packet Option | New NC State | Link-layer Address | Part |
|---|---|---|---|---|
| No | No | REACHABLE | Unchanged | A |
| Same | No | REACHABLE | Unchanged | B |
| Different | No | STALE | Updated | C |
| Different | Yes | STALE | Updated | D |
| Different | Yes, packet > 1280 | STALE | Updated | E |

A. *No TLLA Option, No Redirected Packet Option, Link-layer Address Unchanged:* TR1 and TR2 each transmit a Router Advertisement to the all-nodes multicast address. The Router Advertisements include a Prefix Advertisement with a global prefix and the L and A bits set. This should cause the NUT to add TR1 and TR2 to its Default Router List, configure a global address, and compute Reachable Time. R1 and TR2 each transmit an Echo Request to the NUT and respond to Neighbor Solicitations from the NUT. Wait for Echo Replies from the NUT. This should cause the NUT to resolve the addresses of TR1 and TR2 and create a Neighbor Cache entry for each router in state REACHABLE. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above. Afterwards, TR2 transmits a link-local Echo Request to the HUT.

B. *TLLA Option, No Redirected Packet Option, Link-layer Address Unchanged:* TR1 and TR2 each transmit a Router Advertisement to the all-nodes multicast address. The Router Advertisements include a Prefix Advertisement with a global prefix and the L and A bits set. This should cause the NUT to add TR1 and TR2 to its Default Router List, configure a global address, and compute Reachable Time. R1 and TR2 each transmit an Echo Request to the NUT and respond to Neighbor Solicitations from the NUT. Wait for Echo Replies from the NUT. This should cause the NUT to resolve the addresses of TR1 and TR2 and create a Neighbor Cache entry for each router in state REACHABLE. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above. Afterwards, TR2 transmits a link-local Echo Request to the HUT.

C. *TLLA Option, No Redirected Packet Option, Link-layer Address Updated:* TR1 and TR2 each transmit a Router Advertisement to the all-nodes multicast address. The Router Advertisements include a Prefix Advertisement with a global prefix and the L and A bits set. This should cause the NUT to add TR1 and TR2 to its Default Router List, configure a global address, and compute Reachable Time. R1 and TR2 each transmit an Echo Request to the NUT and respond to Neighbor Solicitations from the NUT. Wait for Echo Replies from the NUT. This should cause the NUT to resolve the addresses of TR1 and TR2 and create a Neighbor Cache entry for each router in state REACHABLE. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above. Afterwards, TR2 transmits a link-local Echo Request to the HUT.

D. *TLLA Option, Redirected Packet Option, Link-layer Address Updated:* TR1 and TR2 each transmit a Router Advertisement to the all-nodes multicast address. The Router Advertisements include a Prefix Advertisement with a global prefix and the L and A bits set. This should cause the NUT to add TR1 and TR2 to its Default Router List, configure a global address, and compute Reachable Time. R1 and TR2 each transmit an Echo Request to the NUT and respond to Neighbor Solicitations from the NUT. Wait for Echo Replies from the NUT. This should cause the NUT to resolve the addresses of TR1 and TR2 and create a Neighbor Cache entry for each router in state REACHABLE. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above. Afterwards, TR2 transmits a link-local Echo Request to the HUT.

E. *TLLA Option, Oversized Redirected Packet Option, Link-layer Address Updated:* TR1 and TR2 each transmit a Router Advertisement to the all-nodes multicast address. The Router Advertisements include a Prefix Advertisement with a global prefix and the L and A bits set. This should cause the NUT to add TR1 and TR2 to its Default Router List, configure a global address, and compute Reachable Time. R1 and TR2 each transmit an Echo Request to the NUT and respond to Neighbor Solicitations from the NUT. Wait for Echo Replies from the NUT. This should cause the NUT to resolve the addresses of TR1 and TR2 and create a Neighbor Cache entry for each router in state REACHABLE. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above. Afterwards, TR2 transmits a link-local Echo Request to the HUT.

**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.13 | **Neighbor Cache Updated from State STALE (Hosts Only)** | **A-E** | **N/A** |

**Purpose**: Verify that a host properly updates its Neighbor Cache entry upon receipt of a valid ICMP Redirect Message.

**Comments on Test Procedure**

| TLLA Option | Redirected Packet Option | New NC State | Link-layer Address | Part |
|---|---|---|---|---|
| No | No | STALE | Unchanged | A |
| Same | No | STALE | Unchanged | B |
| Different | No | STALE | Updated | C |
| Different | Yes | STALE | Updated | D |
| Different | Yes, packet > 1280 | STALE | Updated | E |

A. *No TLLA Option, No Redirected Packet Option, Link-layer Address Unchanged:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR2 transmits an unsolicited Router Advertisement with a Source Link-layer Address option to the all-nodes multicast address. Afterwards, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above. TR2 transmits a link-local Echo Request to the HUT.

B. *TLLA Option, No Redirected Packet Option, Link-layer Address Unchanged:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR2 transmits an unsolicited Router Advertisement with a Source Link-layer Address option to the all-nodes multicast address. Afterwards, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above. TR2 transmits a link-local Echo Request to the HUT.

C. *TLLA Option, No Redirected Packet Option, Link-layer Address Updated:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR2 transmits an unsolicited Router Advertisement with a Source Link-layer Address option to the all-nodes multicast address. Afterwards, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above. TR2 transmits a link-local Echo Request to the HUT.

D. *TLLA Option, Redirected Packet Option, Link-layer Address Updated:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR2 transmits an unsolicited Router Advertisement with a Source Link-layer Address option to the all-nodes multicast address. Afterwards, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above. TR2 transmits a link-local Echo Request to the HUT.

E. *TLLA Option, Oversized Redirected Packet Option, Link-layer Address Updated:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR2 transmits an unsolicited Router Advertisement with a Source Link-layer Address option to the all-nodes multicast address. Afterwards, TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above. TR2 transmits a link-local Echo Request to the HUT.

**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.14 | **Neighbor Cache Updated from State PROBE (Hosts Only)** | **A-E** | **N/A** |

**Purpose**: Verify that a host properly updates its Neighbor Cache entry upon receipt of a valid ICMP Redirect Message.

**Comments on Test Procedure**

| TLLA Option | Redirected Packet Option | New NC State | Link-layer Address | Part |
|---|---|---|---|---|
| No | No | PROBE | Unchanged | A |
| Same | No | PROBE | Unchanged | B |
| Different | No | STALE | Updated | C |
| Different | Yes | STALE | Updated | D |
| Different | Yes, packet > 1280 | STALE | Updated | E |

A. *No TLLA Option, No Redirected Packet Option, Link-layer Address Unchanged:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR2 transmits an unsolicited Router Advertisement with a Source Link-layer Address option to the all-nodes multicast address. Also, TR2 transmits a solicited Neighbor Advertisement for its link-local address to the HUT. Afterwards, TR2 transmits an Echo Request from its link-local address to the HUT. TR1 transmits a Redirect message to the NUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above.

B. *TLLA Option, No Redirected Packet Option, Link-layer Address Unchanged:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR2 transmits an unsolicited Router Advertisement with a Source Link-layer Address option to the all-nodes multicast address. Also, TR2 transmits a solicited Neighbor Advertisement for its link-local address to the HUT. Afterwards, TR2 transmits an Echo Request from its link-local address to the HUT. TR1 transmits a Redirect message to the NUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above.

C. *TLLA Option, No Redirected Packet Option, Link-layer Address Updated:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR2 transmits an unsolicited Router Advertisement with a Source Link-layer Address option to the all-nodes multicast address. Also, TR2 transmits a solicited Neighbor Advertisement for its link-local address to the HUT. Afterwards, TR2 transmits an Echo Request from its link-local address to the HUT. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above.

D. *TLLA Option, Redirected Packet Option, Link-layer Address Updated:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR2 transmits an unsolicited Router Advertisement with a Source Link-layer Address option to the all-nodes multicast address. Also, TR2 transmits a solicited Neighbor Advertisement for its link-local address to the HUT. Afterwards, TR2 transmits an Echo Request from its link-local address to the HUT. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above.

E. *TLLA Option, Oversized Redirected Packet Option, Link-layer Address Updated:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR2 transmits an unsolicited Router Advertisement with a Source Link-layer Address option to the all-nodes multicast address. Also, TR2 transmits a solicited Neighbor Advertisement for its link-local address to the HUT. Afterwards, TR2 transmits an Echo Request from its link-local address to the HUT. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. The Redirect message contains a Target Link-layer Address option or Redirected Packet option according to the table above.

**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.15 | **Invalid Redirect does not Update Neighbor Cache (Hosts Only)** | **A-I** | **N/A** |

**Purpose**: Verify that a host properly processes invalid Redirect messages when redirected on-link.

**Comments on Test Procedure**

A. *Redirect Source Address is Global:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an incorrect IPv6 Source Address (the off-link global address of TN2). After, TR2 transmits a link-local Echo Request to the HUT.

B. *Redirect Source Address is not the current first-hop router:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an incorrect IPv6 Source Address (the off-link global address of TR2). After, TR2 transmits a link-local Echo Request to the HUT.

C. *Hop Limit is not 255:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an incorrect IPv6 Hop Limit of 254. After, TR2 transmits a link-local Echo Request to the HUT.

D. *ICMPv6 Code is not 0:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an incorrect ICMPv6 Code of 1. After, TR2 transmits a link-local Echo Request to the HUT.

E. *ICMPv6 Checksum is invalid:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an incorrect ICMPv6 Checksum. After, TR2 transmits a link-local Echo Request to the HUT.

F. *ICMPv6 Destination Address is Multicast:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an ICMPv6 Destination Address of the all-nodes multicast address. After, TR2 transmits a link-local Echo Request to the HUT.

G. *Target Address is Multicast:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains a Target Address of the All-nodes multicast address. After, TR2 transmits a link-local Echo Request to the HUT.

H. *ICMPv6 length is less than 40 Octets:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an invalid IPv6 Length of 39 bytes. After, TR2 transmits a link-local Echo Request to the HUT.

I. *Option has Length Zero:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Redirect message to the HUT. The ICMPv6 Destination Address is the global address of TN1. The Target Address is the link-local address of TR2. The Redirect message contains an Option with length 0. After, TR2 transmits a link-local Echo Request to the HUT.

**Comments on Test Results**

These tests are performed on Hosts only.



| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.16 | **Redirect – Transmit (Routers Only)** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |

**Purpose**: Verify that a router properly handles transmission of Redirect messages.

**Comments on Test Procedure**

A. *Send Redirect:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an Echo Request to TN2's unicast global address with prefix X and a first hop through the RUT.

B. *Send Redirect to Alternate Router:* TN1 transmits an Echo Request to TN2's unicast global address and a first hop through the RUT.

C. *Source not neighbor:* TN1 transmits an Echo Request to TN2 with a first hop through the RUT. The Source Address is TN1's address with an off-link prefix.

D. *Destination Multicast:* TN1 transmits an Echo Request to TN2's solicited-node multicast address with a first hop through the RUT.

**Comments on Test Results**

A. The RUT transmitted a Redirect message with the following values:

| | |
|---|---|
| **IPv6 Source** | Link-Local address of RUT |
| **IPv6 Destination** | TN1's address (used in Echo Request's Source Address) |
| **Target** | TN2's unicast global address with prefix X |
| **Destination** | TN2's unicast global address with prefix X |
| **TLL Option** | TN2's link-layer address if known |
| **Redirected Header** | TN1's Echo Request without total packet exceeding 1280 bytes. |

B. The RUT transmitted a Redirect message with the following values:

| | |
|---|---|
| **IPv6 Source** | Link-Local address of RUT |

| | |
|---|---|
| **IPv6 Destination** | TN1's address (used in Echo Request's Source Address) |
| **Target** | TR1's link-local address |
| **Destination** | TN2's unicast global address |
| **TLL Option** | TR1's link-layer address if known |
| **Redirected Header** | TN1's Echo Request without total packet exceeding 1280 bytes. |

Parts C-D: The RUT did not send a Redirect message.

| Test # | | Result | |
|---|---|---|---|
| v6LC.2.3.17 | **Redirect – Receive (Routers Only)** | **A** | **PASS** |

**Purpose**: Verify that a router properly handles reception of Redirect messages.

**Comments on Test Procedure**

A. TR1 and TR2 each transmit a Router Advertisement to the all-nodes multicast address. The Router Advertisements include a Prefix Advertisement with a global prefix and the L and A bits set. This should cause the NUT to add TR1 and TR2 to its Default Router List, configure a global address, and compute Reachable Time. TR1 and TR2 each transmit an Echo Request to the NUT and respond to Neighbor Solicitations from the NUT. Wait for Echo Replies from the NUT. This should cause the NUT to resolve the addresses of TR1 and TR2 and create a Neighbor Cache entry for each router in state REACHABLE. TR1 then forwards an Echo Request from TN2 to the RUT. The Destination Address is the global address of the RUT. Also, TR1 transmits a Redirect message to the RUT. The ICMPv6 Destination Address is the global address of TN2. The Target Address is the link-local address of TR2. Afterwards, TR1 forwards an Echo Request from TN2 to the RUT. The Destination Address is the global address of the RUT.

**Comments on Test Results**

A. The RUT sent an Echo Reply with a first hop through TR1. The RUT still sent an Echo Reply with a first hop through TR1, indicating the RUT did not change its routing table with information from TR1's Redirect message.

## Section 3: RFC 2462

These tests are designed to verify the readiness of an IPv6 implementation vis-à-vis the IPv6 Stateless Address Autoconfiguration specification.

## Group 1: Address Autoconfiguration and Duplicate Address Detection

The following tests cover Address autoconfiguration and duplicate address detection in IPv6.

| Test # | | Result | |
|---|---|---|---|
| v6LC.3.1.1 | **Address Autoconfiguration and Duplicate Address Detection** | **A** | **PASS** |
| **Purpose**: Verify that a node can properly initialize on a network using address autoconfiguration and communicate with other on-link partners. | | | |
| **Comments on Test Procedure** | | | |
| A. Initialize all the devices on Link B. Allow time for all devices on Link B to perform stateless address autoconfiguration and DAD. Transmit a DAD NS from TN1 with the Target Address set to the NUT's link-local address. | | | |
| **Comments on Test Results** | | | |
| A. The NUT performed DAD on its tentative address for its interface on Link B sending DupAddrDetectTransmits Neighbor Solicitations, every RetransTimer. The NUT also assigned the tentative address to its interface. The NUT did transmit a Solicited NA for its autoconfigured link-local address. | | | |

| Test # | | Result | |
|---|---|---|---|
| v6LC.3.1.2 | **Receiving DAD Neighbor Solicitations and Advertisements** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |

**Purpose**: To verify that a node can properly process neighbor solicitations and advertisements performing Duplicate Address Detection while the node is also performing DAD.

**Comments on Test Procedure**

A. *NUT receives DAD NS (target != NUT):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT. Configure TN1 to transmit DAD Neighbor Solicitation A with the Target Address set to TN1's link-local address. TN1 transmits a NS with the Target Address set to the NUT's link-local address.
B. *NUT receives DAD NS (target == NUT):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT. Configure TN1 to transmit DAD Neighbor Solicitation A with the Target Address set to the NUT's tentative link-local address. TN1 transmits a NS with the same target address used for the first DAD NS.
C. *NUT receives DAD NA (target != NUT):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT. Configure TN1 to transmit DAD Neighbor Solicitation A with the Target Address set to the NUT's tentative link-local address. TN1 transmits a NS with the target address set to the NUT's link-local address.
D. *NUT receives DAD NA (target == NUT):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT. Configure TN1 to transmit DAD Neighbor Solicitation A with the Target Address set to the NUT's tentative link-local address. TN1 transmits a NS with the same target address used for the first DAD NS.

**Comments on Test Results**

A. The NUT silently ignored the DAD NS. The NUT completed the DAD process and assign the tentative address to its interface. The NUT did transmit a Solicited NA for its autoconfigured link-local address.
B. The NUT received more DAD NS messages than expected with its tentative link-local address as the Target address. The NUT then determined its tentative address is a duplicate and should not assign the tentative address to its interface. The NUT did not transmit a Solicited NA for its autoconfigured link-local address.
C. The NUT silently ignored DAD NA. The NUT completed the DAD process and assign the tentative address to its interface. The NUT did transmit a Solicited NA for its autoconfigured link-local address.
D. The NUT determined its tentative address is not unique and did not assign the tentative address to its interface. The NUT did not transmit a Solicited NA for its autoconfigured link-local address.

| Test # | | Result | |
|---|---|---|---|
| v6LC.3.1.3 | **Validation of DAD Neighbor Solicitations** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **FAIL** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |
| | | **G** | **PASS** |
| | | **H** | **FAIL** |
| | | **I** | **PASS** |
| | | **J** | **PASS** |

**Purpose**: Verify that a node can properly initialize on a network using address autoconfiguration and communicate with other on-link partners.

**Comments on Test Procedure**

A. *NUT receives invalid DAD NS (ICMP length < 24 octets):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Solicitation A with the ICMP length set to 16. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

B. *NUT receives invalid DAD NS (HopLimit !=255):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Solicitation A with the Hoplimit set to 254. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

C. *NUT receives invalid DAD NS (Dst = NUT's tentative address):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Solicitation A with the Destination address set to the NUT's tentative link-local address. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

D. *NUT receives invalid DAD NS (Dst = allnode):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Solicitation A with Destination address set to the all-nodes multicast address. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

E. *NUT receives invalid DAD NS (ICMP code!= zero):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Solicitation A with the ICMP code set to 1. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

F. *NUT receives invalid DAD NS (Invalid Checksum):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Solicitation A with an invalid ICMP Checksum. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

G. *NUT receives invalid DAD NS (target == multicast address):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Solicitation A with the Target Address set to the solicited multicast of the NUT's tentative link-local address. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

H. *NUT receives invalid DAD NS (contains SLL):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Solicitation A containing a SLL Option set to TN1's MAC address. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

I. *NUT receives valid DAD NS (Reserved Field):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Solicitation A with the Reserved field set to 0xFFFFFFFF. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.
J. *NUT receives valid DAD NS (contains TLL):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Solicitation A containing a TLL Option set to TN1's MAC address. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

**Comments on Test Results**

A. The NUT silently ignored the invalid DAD NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured link-local address.
B. The NUT silently ignored the invalid DAD NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured link-local address.
C. The NUT silently ignored the invalid DAD NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured link-local address.
D. The NUT did not silently ignore the invalid DAD NS. The NUT did not transmit a Solicited NA for its autoconfigured link-local address.

   According to RFC 2462, Section 5.4.1: "A node MUST silently discard any Neighbor Solicitation or Advertisement message that does bit pass the validity checks specified in [DISCOVERY]."

   Therefore, the NUT should have silently discarded the invalid DAD Neighbor Solicitation and continued with the DAD process before transmitting a Solicited Neighbor Advertisement for its autoconfigured link-local address.

E. The NUT silently ignored the DAD NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured link-local address.
F. The NUT silently ignored the DAD NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured link-local address.
G. The NUT silently ignored the invalid DAD NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured link-local address.
H. The NUT did not silently ignore the invalid DAD NS. The NUT did not transmit a Solicited NA for its autoconfigured link-local address.

   According to RFC 2462, Section 5.4.1: "A node MUST silently discard any Neighbor Solicitation or Advertisement message that does bit pass the validity checks specified in [DISCOVERY]."

   Therefore, the NUT should have silently discarded the invalid DAD Neighbor Solicitation and continued with the DAD process before transmitting a Solicited Neighbor Advertisement for its autoconfigured link-local address.

I. The NUT ignored the contents of the Reserved field. The NUT did not assign the tentative address to its interface. The NUT did not transmit a Solicited NA for its autoconfigured link-local address.
J. The NUT ignored any options they do not recognize and continued processing the message. The NUT also did not assign the tentative address to its interface. The NUT did not transmit a Solicited NA for its autoconfigured link-local address.

| Test # | | Result | |
|---|---|---|---|
| v6LC.3.1.4 | **Receiving Invalid Neighbor Advertisements** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |
| | | **G** | **PASS** |
| | | **H** | **PASS** |
| | | **I** | **PASS** |

**Purpose**: Verify that a node can properly ignore invalid neighbor advertisements while performing Duplicate Address Detection.

**Comments on Test Procedure**

A. *NUT receives invalid DAD NA (ICMP length < 24 octets):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Advertisement A with the ICMP length set to 16. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

B. *NUT receives invalid DAD NA (HopLimit != 255):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Advertisement A with the Hoplimit set to 254. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

C. *NUT receives invalid DAD NA (ICMP code!= zero):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Advertisement A with the ICMP code set to 1. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

D. *NUT receives invalid DAD NA (Invalid Checksum):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Advertisement A with an invalid ICMP Checksum. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

E. *NUT receives invalid DAD NA (SolicitedFlag ==1):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Advertisement A with Solicited flag set to 1. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

F. *NUT receives invalid DAD NA (target == multicast address):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Advertisement A with the Target Address set to the solicited multicast of the NUT's tentative link-local address. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

G. *NUT receives invalid DAD NA (option length ==zero):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Advertisement A with the TLLOPT Length set to 0. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

H. *NUT receives valid DAD NA (Reserved Field):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Advertisement A with the Reserved field set to 0x1FFFFFFF. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

I. *NUT receives valid DAD NA (contains SLL):* First, initialize all devices on Link B. After TN1 receives a DAD NS message from the NUT, configure TN1 to transmit Neighbor Advertisement A containing a SLL Option set to TN1's MAC address. TN1 transmits a valid Solicited NS with the target address set to the NUT's link-local address.

**Comments on Test Results**

A. The NUT silently ignored the invalid DAD NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured Link-Local Address.
B. The NUT silently ignored the invalid DAD NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured Link-Local Address.
C. The NUT silently ignored the invalid DAD NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured Link-Local Address.
D. The NUT silently ignored the invalid DAD NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured Link-Local Address.
E. The NUT silently ignored the DAD NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured Link-Local Address.
F. The NUT silently ignored the DAD NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured Link-Local Address.
G. The NUT silently ignored the invalid DAD NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured Link-Local Address.
H. The NUT ignored the contents of the Reserved field. The NUT did not assign the tentative address to its interface. The NUT also did not transmit a Solicited NA for its autoconfigured Link-Local Address.
I. The NUT ignored any options they do not recognize and continue processing the message. The NUT did not assign the tentative address to its interface. The NUT also did not transmit a Solicited NA for its autoconfigured Link-Local Address.



| Test # | | Result | |
|---|---|---|---|
| v6LC.3.1.5 | **Receiving Neighbor Solicitations for Address Resolution** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a node can properly ignore neighbor solicitations performing address resolution while performing Duplicate Address Detection.

**Comments on Test Procedure**

A. *NUT receives NS (src == unicast):* First, initialize all the devices on Link B. After TN1 receives a DAD NS message from the NUT. Configure TN1 to transmit Neighbor Solicitation A. TN1 transmits a DAD NS with the Target Address set to the NUT's Link-Local Address.
B. *NUT receives NS (Src == unicast && Dst == NUT's tentative address):* First, initialize all the devices on Link B. After TN1 receives a DAD NS message from the NUT. Configure TN1 to transmit Neighbor Solicitation A with the Destination Address set to the NUT's tentative Link-Local Address. TN1 transmits a DAD NS with the Target Address set to the NUT's Link-Local Address.

**Comments on Test Results**

A. The NUT silently ignored the NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured Link-Local Address.
B. The NUT silently ignored the NS. The NUT did complete the DAD process and assign the tentative address to its interface. The NUT also transmitted a Solicited NA for its autoconfigured Link-Local Address.

## Group 2: Router Advertisement Processing and Address Lifetime

The following tests cover Router Advertisement processing and address lifetime expiry in IPv6.

| Test # | | Result | |
|---|---|---|---|
| v6LC.3.2.1 | **Address Lifetime Expiry (Hosts Only)** | **A** | **N/A** |
| **Purpose**: Verify that a host can properly handle expired or invalid addresses. | | | |
| **Comments on Test Procedure** | | | |
| A. First, initialize all devices on Link B. TR1 sends out ONE Router Advertisement on Link B with Prefix "X" with a valid lifetime set to 40 seconds. TR1 transmits a NS message for address resolution with the target address set to the NUT's Global Address for Prefix "X". After waiting 35 seconds, TR1 transmits again a NS message for address resolution with the target address set to the NUT's Global Address for Prefix "X". After waiting 10 seconds, TR1 transmits a NS message for address resolution with the target address set to the NUT's Global Address for Prefix "X". | | | |
| **Comments on Test Results** | | | |
| A. This test is performed on Hosts only. | | | |

| Test # | | Result | |
|---|---|---|---|
| v6LC.3.2.2 | **Multiple Prefixes and Network Renumbering (Hosts only)** | **A** | **N/A** |
| **Purpose**: To verify that a host configured with multiple prefixes can communicate with another host on a different network when its site has been renumbered. | | | |
| **Comments on Test Procedure** | | | |
| A. Configure TR1 to discontinue to send RA's for Prefix "X" and also configure TR1 to send out Router Advertisements on Link B with Prefix "Y" with a Valid Lifetime of 30 seconds. Then, configure TR1 to transmit a NS message for address resolution with the Target Address set to the HUT's Global Address for Prefix "X" and "Y". After waiting 11 seconds for Prefix "X" to timeout, configure TR1 to transmit a NS message for address resolution with the Target Address set to the HUT's global address for Prefix "X" and "Y". Afterwards, configure TR1 to transmit a NS message for address resolution with the target address set to the HUT's Global Address for Prefix "Y". | | | |
| **Comments on Test Results** | | | |
| A. This test is performed on Hosts only. | | | |

| Test # | | Result | |
|---|---|---|---|
| v6LC.3.2.3 | **Prefix-Information Option Processing (Hosts Only)** | **A-I** | **N/A** |

**Purpose**: Verify that a host properly processes the Prefix Information Option in the Router Advertisement

**Comments on Test Procedure**

A. *Router Advertisement with multiple Prefix Options:* First, initialize all devices on Link B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Router Advertisement with the Autonomous flag set, NextHop=255, and multiple prefix options, Prefix "X" with a valid lifetime of 20s and Prefix "Y" with a valid lifetime of 40s. After, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X" and "Y". After waiting 21 seconds, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X". After waiting 20 seconds, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "Y".

B. *Autonomous Flag not set:* First, initialize all devices on Link B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a Router Advertisement A with the Autonomous flag not set. Afterwards, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X".

C. *Prefix is set to link-local prefix:* First, initialize all devices on Link B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits Router Advertisement A with the prefix set the link-local prefix. Afterwards, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X".

D. *Preferred lifetime > valid lifetime:* First, initialize all devices on Link B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits Router Advertisement A with the preferred lifetime set to 30 seconds. Afterwards, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X".

E. *Prefix length > 128 bits:* First, initialize all devices on Link B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits Router Advertisement A with a Prefix Length set to 128. Afterwards, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X".

F. *Prefix length < 64 bits:* First, initialize all devices on Link B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits Router Advertisement A with a Prefix Length set to zero. Afterwards, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X".

G. *(64 bits < prefix length < 128 bits):* First, initialize all devices on Link B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits Router Advertisement A with a Prefix Length set to 120. Afterwards, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X".

H. *Valid Lifetime is zero:* First, initialize all devices on Link B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits Router Advertisement A with the Valid Lifetime set to zero. Afterwards, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X".

I. *Invalid RA with Hop Limit 254:* First, initialize all devices on Link B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits Router Advertisement A with a Hop Limit set to 254. Afterwards, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X".

**Comments on Test Results**

These tests are performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.3.2.4 | **Prefix-Information Option Processing, Lifetime (Hosts Only)** | **A-D** | **N/A** |

**Purpose**: Verify that a host properly updates its Address List upon receipt of Prefix Information Options.

**Comments on Test Procedure**

A. *Prefix Lifetime greater than Stored Lifetime*: First, initialize the HUT on Link B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits Router Advertisement A with a Valid Lifetime of 30 seconds. After waiting 10 seconds, TR1 transmits a Router Advertisement with a prefix of TR1's Global Prefix and a Valid Lifetime of 60 seconds. After waiting 25 seconds, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X".

B. *Prefix Lifetime greater than 2 hours:* First, initialize the HUT on Link B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits Router Advertisement A with a Valid Lifetime of 3hrs. After, TR1 transmits a Router Advertisement with a prefix of TR1's Global Prefix and a Valid Lifetime of 2hrs 30s. After waiting two hours and 45 seconds, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X".

C. *Prefix Lifetime less than the Stored Lifetime and the Stored Lifetime is less than 2 hours:* First, initialize the HUT on Link B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits Router Advertisement A with its L flag set and its retransmit interval set to 1 second. Following the last RA, TR1 transmits a Router Advertisement with a prefix of TR1's Global Prefix and a Valid Lifetime of 30 seconds. After waiting 35 seconds, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X".

D. *Prefix Lifetime less than 2 hours and the Stored Lifetime is greater than 2 hours:* First, initialize the HUT on Link B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits Router Advertisement A with a Valid Lifetime of 2hrs 30s. Following the last RA, TR1 transmits a Router Advertisement with a prefix of TR1's Global Prefix and a Valid Lifetime of 10 seconds. After waiting 11 seconds, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X". After waiting another two hours and 15 seconds, TR1 transmits a NS message for address resolution with the target address set to the HUT's global address for Prefix "X".



**Comments on Test Results**

These tests are performed on Hosts only.

## Section 4: RFC 1981

These tests are designed to verify the readiness of an IPv6 implementation vis-à-vis the Path MTU Discovery IPv6 specification.

| Test # | | Result | |
|---|---|---|---|
| v6LC.4.1.1 | **Confirm Ping** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |

**Purpose**: Verify that a node can reply to variable sized ICMP Echo Requests.

**Comments on Test Procedure**

A. *ICMPv6 Echo Request 64 octets:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN2 to the NUT. The packet size of the Echo Request is 64 octets.
B. *ICMPv6 Echo Request 1280 octets:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN2 to the NUT. The packet size of the Echo Request is 1280 octets.
C. *ICMPv6 Echo Request 1500 octets:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN2 to the NUT. The packet size of the Echo Request is 1500 octets.

**Comments on Test Results**

A. The NUT sent an Echo Reply to TR1 64 octets in packet size.
B. The NUT sent an Echo Reply to TR1 1280 octets in packet size.
C. The NUT sent an Echo Reply to TR1 1500 octets in packet size.



| Test # | | Result | |
|---|---|---|---|
| v6LC.4.1.2 | **Stored PMTU** | **A** | **PASS** |

**Purpose**: Verify that a node can store Path MTU information for multiple destinations.

**Comments on Test Procedure**

A. TN1 sends an Echo Request on-link to the NUT with packet size equal to 1500 octets. TR1 forwards an Echo Request from TN2 to the NUT with packet size equal to 1500 octets. Afterwards, TR1 forwards an Echo Request from TN3 to the NUT with packet size equal to 1500 octets. TR1 then transmits a Packet Too Big message to the NUT for the Echo Reply to TN2, which contains an MTU field with a value of 1400. TN1 sends an Echo Request on-link to the NUT with packet size equal to 1500 octets. TR1 forwards an Echo Request from TN2 to the NUT with packet size equal to 1500 octets. Afterwards, TR1 forwards an Echo Request from TN3 to the NUT with packet size equal to 1500 octets. TR1 transmits another Packet Too Big message to the NUT for the Echo Reply to TN3, which contains an MTU field with a value of 1280. TN1 sends an Echo Request on-link to the NUT with packet size equal to 1500 octets. TR1 forwards an Echo Request from TN2 to the NUT with packet size equal to 1500 octets. Afterwards, TR1 forwards an Echo Request from TN3 to the NUT with packet size equal to 1500 octets.

**Comments on Test Results**

A. The NUT sent three Echo Replies, one to TN1, one to TN2, and one to TN3. The NUT then responded to the three Echo Requests. The Echo Replies to TN1 and TN3 were no larger than 1500 octets. The NUT also did not have to fragment these packets. The NUT correctly fragmented its Echo Reply to TN2 with each fragment no larger than 1400 octets. The NUT again responded to the three Echo Requests. The Echo Reply to TN1 should be no larger than 1500 octets. The NUT did not have to fragment this packet. The NUT correctly fragmented its Echo Reply to TN2 with each fragment no larger than 1400 octets. After, the NUT correctly fragmented its Echo Reply to TN3 with each fragment no larger than 1280 octets.

| Test # | | Result | |
|---|---|---|---|
| v6LC.4.1.3 | **Non-zero ICMPv6 Code** | **A** | **PASS** |
| **Purpose**: Verify that a node properly processes a Packet Too Big message with a non-zero ICMPv6 Code field. | | | |
| **Comments on Test Procedure** | | | |
| A. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN2 to the NUT. Also, TR1 transmits a Packet Too Big message to the NUT, which contains an invalid ICMPv6 Code field value of 0xFF. The MTU field is set to 1280. Afterwards, TR1 forwards an Echo Request from TN2 to the NUT. | | | |
| **Comments on Test Results** | | | |
| A. The NUT responded to the Echo Request using TR1 as a first hop. The NUT did correctly fragment its response to the Echo Request using TR1 as a first hop, indicating the NUT ignored the invalid ICMPv6 Code field and processed the Packet Too Big message. The fragmented packets were not larger than 1280 octets in size. | | | |



| Test # | | Result | |
|---|---|---|---|
| v6LC.4.1.4 | **Reduce PMTU On-link** | **A** | **PASS** |
| **Purpose**: Verify that a node properly processes a Packet Too Big message indicating a reduction in Path MTU for a link-local destination. | | | |
| **Comments on Test Procedure** | | | |
| A. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 transmits a link-local Echo Request to the NUT. Afterwards, TR1 transmits a Packet Too Big message to the NUT with an MTU of 1280. Following the Too Big Message, TR1 transmits another link-local Echo Request to the NUT. | | | |
| **Comments on Test Results** | | | |
| A. The NUT responded to the Echo Request. The NUT did correctly fragment its response to the Echo Request, indicating the NUT processed the Packet Too Big message. The fragmented packets were no larger than 1280 octets in size. | | | |

| Test # | | Result | |
|---|---|---|---|
| v6LC.4.1.5 | **Reduce PMTU Off-link** | **A** | **PASS** |

**Purpose**: Verify that a node properly processes a Packet Too Big message indicating a reduction in Path MTU for a link-local destination.

**Comments on Test Procedure**

A. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN2 to the NUT. Afterwards, TR1 transmits a Packet Too Big message to the NUT with an MTU field set to 1400 octets. TR1 forwards an Echo Request from TN2 to the NUT with a packet size of 1500 octets. TR1 transmits another Packet Too Big message containing an MTU field set to 1280 octets followed by, TR1 forwarding an Echo Request from TN2 to the NUT with a packet size of 1500 octets.

**Comments on Test Results**

A. The NUT responded to the Request using TR1 as the first hop. The NUT did correctly fragment its response to the Echo Request using TR1 as a first hop, indicating the NUT processed the Packet Too Big message. The fragmented packets were no larger than 1400 octets in size. Again, The NUT did correctly fragment its response to the Echo Request using TR1 as a first hop, indicating the NUT processed the Packet Too Big message. The fragmented packets were no larger than 1280 octets in size.



| Test # | | Result | |
|---|---|---|---|
| v6LC.4.1.6 | **Receiving MTU Below IPv6 Minimum Link MTU** | **A** | **FAIL** |
| | | **B** | **FAIL** |

**Purpose**: Verify that a node does not reduce its estimate of the Path MTU below the IPv6 minimum link MTU.

**Comments on Test Procedure**

A. *MTU equal to 0x0:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN2 to the NUT. The packet size is 1280 octets. Also, TR1 transmits a Packet Too Big message to the NUT, which contains an MTU field of 0x0. Afterwards, TR1 forwards an Echo Request from TN2 to the NUT. The packet size is 1280 octets.

B. *MTU equal to 512:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN2 to the NUT. The packet size is 1280 octets. Also, TR1 transmits a Packet Too Big message to the NUT, which contains an MTU field 512 octets. Afterwards, TR1 forwards an Echo Request from TN2 to the NUT. The packet size is 1280 octets.

**Comments on Test Results**

Parts A,B. The NUT responded to the second Echo Request without including a Fragment header in the Echo Reply Packet despite receiving a Packet Too Big message from TR1.

According to RFC 2460, Section 5: "[…] The originating IPv6 node may receive a Packet Too Big message reporting a Next-Hop MTU less than 1280. In that case, the IPv6 node is not required to reduce the size of the subsequent packets to less than 1280, but MUST include a Fragment header in those packets […]".

Therefore, the NUT should have responded to the second Echo Request with an Echo Reply packet that included a Fragment header after receiving the Packet Too Big message fromTR1.

| Test # | | Result | |
|---|---|---|---|
| v6LC.4.1.7 | **Increase Estimate** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a node does not increase its estimate of the MTU for a path due to a Packet Too Big message.

**Comments on Test Procedure**

A. *MTU increase:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN2 to the NUT with packet size equal to 1500 octets. TR1then transmits a Packet Too Big message to the NUT. The MTU field is 1304 octets. Afterwards, TR1 forwards an Echo Request from TN2 to the NUT with packet size equal to 1500 octets. TR1 transmits another Packet Too Big message to the NUT. The MTU field is 1500 octets. Afterwards, TR1 forwards an Echo Request from TN2 to the NUT. The packet size is 1500 octets.

B. *MTU equal to 0x1FFFFFFF:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN2 to the NUT with packet size equal to 1500 octets. TR1then transmits a Packet Too Big message to the NUT. The MTU field is 1304 octets. Afterwards, TR1 forwards an Echo Request from TN2 to the NUT with packet size equal to 1500 octets. TR1 transmits another Packet Too Big message to the NUT. The MTU field of 0x1FFFFFFF. Afterwards, TR1 forwards an Echo Request from TN2 to the NUT. The packet size is 1500 octets.

**Comments on Test Results**

Parts A-B: The NUT responded to the Echo Request using TR1 as a first hop. The NUT did fragment the response to the Echo Request using TR1 as a first hop, indicating the NUT processed the Packet Too Big message. The NUT also correctly fragmented the response to the Echo Request using TR1 as a first hop so the packet size is equal to or under 1304 octets. The NUT did not process the second Packet Too Big message indicating an increase in the PMTU.



| Test # | | Result | |
|---|---|---|---|
| v6LC.4.1.8 | **Router Advertisement with MTU Option (Hosts Only)** | **A** | **N/A** |

**Purpose**: Verify that a host properly processes a Router Advertisement with an MTU option.

**Comments on Test Procedure**

A. *Reduce estimate due to MTU option:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TR1 forwards an Echo Request from TN2 to the HUT with packet size equal to 1500 octets. TR1 then transmits another Router Advertisement with an MTU option set to 1280 to the all-nodes multicast address. Afterwards, TR1 forwards a fragmented Echo Request from TN2 to the HUT with reassembled packet size equal to 1500 octets.

**Comments on Test Results**

A. This test is performed on Hosts only.

| Test # | | Result | |
|---|---|---|---|
| v6LC.4.1.9 | **Checking For Increase in PMTU** | **A** | **PASS** |

**Purpose**: Verify that a node waits the proper amount of time to check for PMTU increases.

**Comments on Test Procedure**

A. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. Afterwards, TR1 forwards an Echo Request from TN2 to the NUT. TR1 also transmits a Packet Too Big message to the NUT and TR1 forwards an Echo Request from TN2 to the NUT. The MTU field of the Too Big Message is 1304 octets. Also, TR1 forwards an Echo Request from TN2 every 30 seconds for 5 minutes after the Packet Too Big Message was sent.

**Comments on Test Results**

A. The NUT responded to the Echo Request. The NUT did correctly fragment the response to the Echo Request, indicating it processed the Packet Too Big Message from TR1. The fragmented packets were no larger than 1304 octets in size. The NUT did not transmit any packets larger than 1304 octets for 5 minutes from the time it received the Packet Too Big Message from TR1.



| Test # | | Result | |
|---|---|---|---|
| v6LC.4.1.10 | **Multicast Destination – One Router** | **A** | **N/T** |

**Purpose**: Verify that a node properly chooses the PMTU for multicast destinations.

**Comments on Test Procedure**

A. Transmit an ICMPv6 Echo Request from the NUT with packet size equal to 1500 octets and a destination to the multicast address of FF1E::1:2. TR1 transmits a Packet Too Big Message to the NUT including an MTU field of 1450. Again, transmit an ICMPv6 Echo Request from the NUT with packet size equal to 1500 octets and a destination to the multicast address of FF1E::1:2. TR1 transmits a Packet Too Big Message to the NUT including an MTU field of 1400. Once again, transmit an ICMPv6 Echo Request from the NUT with packet size equal to 1500 octets and a destination to the multicast address of FF1E::1:2. TR1 transmits a Packet Too Big Message to the NUT including an MTU field of 1300. Now, transmit an ICMPv6 Echo Request from the NUT with packet size equal to 1400 octets and a destination to the multicast address of FF1E::1:2. TR1 transmits a Packet Too Big Message to the NUT including an MTU field of 1350. For the last packet to be sent, transmit an ICMPv6 Echo Request from the NUT with packet size equal to 1400 octets and a destination to the multicast address of FF1E::1:2.

**Comments on Test Results**

A. When the NUT transmitted an Echo Request to FF1E::1:2, it automatically fragmented the packets sent. The NUT, despite being configured on the appropriate interfaces with an MTU greater than 1280, continued to fragment its packets. The test was not able to be completed as a result.

| Test # | | Result | |
|---|---|---|---|
| v6LC.4.1.11 | **Multicast Destination – Two Router** | **A** | **N/T** |

**Purpose**: Verify that a node properly chooses the PMTU for multicast destinations when receiving PTB messages from more than one router.

**Comments on Test Procedure**

A. Transmit an ICMPv6 Echo Request from the NUT with packet size equal to 1500 octets and a destination to the multicast address of FF1E::1:2. TR1 transmits a Packet Too Big Message to the NUT including an MTU field of 1480. Again, transmit an ICMPv6 Echo Request from the NUT with packet size equal to 1500 octets and a destination to the multicast address of FF1E::1:2. TR1 transmits a Packet Too Big Message to the NUT including an MTU field of 1440. Once again, transmit an ICMPv6 Echo Request from the NUT with packet size equal to 1500 octets and a destination to the multicast address of FF1E::1:2. TR1 transmits two Packet Too Big Messages to the NUT including an MTU field of 1400 and one of 1360. Again, transmit an ICMPv6 Echo Request from the NUT with packet size equal to 1500 octets and a destination to the multicast address of FF1E::1:2. TR1 transmits two Packet Too Big Messages to the NUT including an MTU field of 1280 and of 1360. For the last packet to be sent, transmit an ICMPv6 Echo Request from the NUT with packet size equal to 1500 octets and a destination to the multicast address of FF1E::1:2.

**Comments on Test Results**

A. When the NUT transmitted an Echo Request to FF1E::1:2, it automatically fragmented the packets sent. The NUT, despite being configured on the appropriate interfaces with an MTU greater than 1280, continued to fragment its packets. The test was not able to be completed as a result.

## Section 5: RFC 2463

These tests are designed to verify conformance with the Internet Control Message Protocol for the Internet Protocol Version 6 Specification.

| Test # | | Result | |
|---|---|---|---|
| v6LC.5.1.1 | **Transmitting Echo Requests** | **A** | **PASS** |
| **Purpose**: Verify that a node properly transmits ICMPv6 Echo Requests. | | | |
| **Comments on Test Procedure** | | | |
| A. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. Use Ping (or any available application for sending Echo Requests) to send an Echo Request from the NUT to TN1's Link-Local address. | | | |
| **Comments on Test Results** | | | |
| A. The NUT sent an Echo Request to TN1. The Destination Address of the Packet was be same as TN1's Link-Local Address. The checksum was also be valid. The Type field was equal to 128 and the Code field was equal to 0. | | | |

| Test # | | Result | |
|---|---|---|---|
| v6LC.5.1.2 | **Replying to Echo Requests** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| **Purpose**: Verify that a node properly replies to ICMPv6 Echo Requests. | | | |
| **Comments on Test Procedure** | | | |
| A. *Request sent to Link-Local address:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an ICMPv6 Echo Request to the NUT's Link-Local address. The source address is TN1's Link-Local address.<br>B. *Request sent to global address:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an ICMPv6 Echo Request to the NUT's Global Address. The source address is TN1's Global Address.<br>C. *Request sent to multicast address:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an ICMPv6 Echo Request to the All-Nodes Link-Local Scope Multicast address (FF02::1). The source address is TN1's Link-Local address. | | | |
| **Comments on Test Results** | | | |
| A. The NUT sent an Echo Reply to TN1. The Source Address of the Packet was the same as the Link-Local Destination Address of TN1's Echo Request packet, while the Destination Address was the same as the Link-Local Source Address of TN1's Echo Request packet. The NUT sent an Echo Reply to TN1 with a valid checksum. | | | |

B. The NUT sent an Echo Reply to TN1. The Source Address of the Packet was the same as the Global Destination Address of TN1's Echo Request packet, while the Destination Address was the same as the Global Source Address of TN1's Echo Request packet. The NUT sent an Echo Reply to TN1 with a valid checksum.
C. The NUT sent an Echo Reply to TN1. The Source Address of the Packet was one of the NUT's unicast addresses belonging to the interface on which the Echo Request was received. This was either a Link-Local or Global address. The Destination Address was TN1's local address Echo Request packet. The NUT sent an Echo Reply to TN1 with a valid checksum.

| Test # | | Result | |
|---|---|---|---|
| v6LC.5.1.3 | **Destination Unreachable Message Generation** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |

**Purpose**: Verify that a node properly generates Destination Unreachable Messages.

**Comments on Test Procedure**

A. *Route Unreachable – Routers Only:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an ICMPv6 Echo Request to an off-link address with a prefix that does not exist.
B. *Address Unreachable – Routers Only:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an ICMPv6 Echo Request to an on-link address that does not exist. The prefix should be set to the prefix assigned by the RUT.
C. *Port Unreachable – Link-Local Address – All Nodes:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits a UDP Packet with the destination port field set to 9000. The source address is TN1's Link-Local address.
D. *Port Unreachable – Global Address – All Nodes:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits a UDP Packet with the destination port field set to 9000. The source address is TN1's Global address.

**Comments on Test Results**

A. The RUT sent a Destination Unreachable Message to TN1. The Source Address of the Packet was one of the RUT's unicast addresses, while the Destination Address was the same as the Source Address in TN1's Echo Request packet.
B. The RUT sent a Destination Unreachable Message to TN1. The Source Address of the Packet was one of the RUT's unicast addresses, while the Destination Address was the same as the Source Address in TN1's Echo Request packet. The Code field was set to "3". The invoking Echo Request packet included in the Error Message did not exceed minimum IPv6 MTU.
C. The NUT sent a Destination Unreachable Message to TN1. The Source Address of the Packet was one of the NUT's unicast addresses, while the Destination Address was the same as the Link-Local Source Address in TN1's packet. The Code field was set to "4". The invoking Echo Request packet included in the Error Message did not exceed minimum IPv6 MTU.

D. The NUT sent a Destination Unreachable Message to TN1. The Source Address of the Packet was one of the NUT's unicast addresses, while the Destination Address was the same as the Global Source Address in TN1's packet. The Code field was set to "4". The invoking Echo Request packet included in the Error Message did not exceed minimum IPv6 MTU.

| Test # | | Result | |
|---|---|---|---|
| v6LC.5.1.4 | **Packet Too Big Message Generation (Routers Only)** | **A** | **PASS** |
| | | **B** | **N/T** |

**Purpose**: Verify that a router properly generates Packet Too Big Messages.

**Comments on Test Procedure**

A. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an Echo Request to TN2 using the RUT as the first-hop with a packet size of 1500 octets.
B. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an Echo Request to FF1E::1:2 using the RUT as the first-hop with a packet size of 1500 octets.

**Comments on Test Results**

A. The RUT did transmit a Packet Too Big message to TN1, as it could not forward the Echo Request due to PMTU limitations.
- The MTU field of Packet Too Big Message was set to 1280.
- The Source Address of the Packet was one of the RUT's unicast addresses for its interface to Link A (to TN2).
- The Destination Address was the same as the Source Address in TN1's Echo Request packet. The Code field was set to "0".
- The invoking Echo Request packet included in the Error Message did not exceed minimum IPv6 MTU.

B. This test was not performed due to configuring a multicast routing protocol.

| Test # | | Result | |
|---|---|---|---|
| v6LC.5.1.5 | **Hop Limit Exceeded (Time Exceeded Generation) (Routers Only)** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a router properly generates Time Exceeded Messages the Hop Limit was exceeded in transit.

**Comments on Test Procedure**

A. *Receive Hop Limit 0:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits the Packet A Echo Request to TN2 with a first hop of the RUT.
B. *Decrement Hop Limit to 0:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits the Packet B Echo Request to TN2 with a first hop of the RUT.

**Comments on Test Results**

A. The RUT discarded the ICMPv6 Echo Request from TN1. Therefore, it did not forward the Echo Request to TN2. The RUT sent a Time Exceeded Message to TN1 with a code field value of 0
B. The RUT discarded the ICMPv6 Echo Request from TN1. Therefore, it did not forward the Echo Request to TN2. The RUT decremented the Hop Limit to 0 and sent a Time Exceeded Message to TN1 with a code field value of 0



| Test # | | Result | |
|---|---|---|---|
| v6LC.5.1.6 | **Erroneous Header Field (Parameter Problem Generation)** | **A** | **PASS** |

**Purpose**: Verify that a node properly generates Parameter Problem Messages for an Erroneous Header Field.

**Comments on Test Procedure**

A. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits the Packet A Echo Request to the NUT. The Source Address of the Packet is set to TN1's Global address. The Destination Address of the packet is set to the NUT's Global address.

**Comments on Test Results**

A. The NUT discarded the ICMPv6 Echo Request from TN1. Therefore, it did not send an Echo Reply. The NUT sent a Parameter Problem Message to TN1 with a code field value of 0 (Erroneous Header Field encountered) because the Payload Length is not a multiple of 8 octets.
   - The Pointer Field was be 0x04 (offset of the Payload Length field).
   - The Source Address of the Packet was the same as the Global Destination Address of TN1's Echo Request packet.
   - The Destination Address was the same as the Global Source Address of TN1's Echo Request packet.
   - The invoking Echo Request packet included in the Error Message did not exceed minimum IPv6 MTU.

| Test # | | Result | |
|---|---|---|---|
| v6LC.5.1.7 | **Unrecognized Next Header (Parameter Problem Generation)** | **A** | **PASS** |

**Purpose**: Verify that a node properly generates Parameter Problem Messages when an Unrecognized Next Header type is encountered.

**Comments on Test Procedure**

A. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits the Packet A Echo Request to the NUT. The Source Address of the Packet is set to TN1's Global address. The Destination Address of the packet is set to the NUT's Global address.

**Comments on Test Results**

A. The NUT discarded the ICMPv6 Echo Request from TN1. Therefore, it did not send an Echo Reply. The NUT sent a Parameter Problem Message to TN1 with a code field value of 1 (Unrecognized Next Header type encountered).
   - The Pointer Field was 0x28 (offset of the Next Header field).
   - The Source Address of the Packet was the same as the Global Destination Address of TN1's Echo Request packet.
   - The Destination Address was the same as the Global Source Address of TN1's Echo Request packet.
   - The invoking Echo Request packet included in the Error Message did not exceed minimum IPv6 MTU.



| Test # | | Result | |
|---|---|---|---|
| v6LC.5.1.8 | **Unknown Informational Message Type** | **A** | **PASS** |

**Purpose**: Verify that a node properly handles the reception of an ICMPv6 Packet with an Unknown Informational Message Type value.

**Comments on Test Procedure**

A. TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an ICMPv6 Information Message with a type field value of 255 to the NUT.

**Comments on Test Results**

A. The NUT silently discarded the ICMPv6 Informational Message from TN1.

| Test # | | Result | |
|---|---|---|---|
| v6LC.5.1.9 | **Error Condition With ICMPv6 Error Message (Routers Only)** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |
| | | **E** | **PASS** |
| | | **F** | **PASS** |

**Purpose**: Verify that a router properly handles the reception and processing of an ICMPv6 Error Message that invokes an error.

**Comments on Test Procedure**

A. *Reception of Flawed Destination Unreachable Code 0 with Address Unreachable*: TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits a Destination Unreachable Error Message for "No Route To Destination" to the RUT with the Destination Address set to an on-link address that does not exist.

B. *Reception of Flawed Destination Unreachable Code 3 with Hop Limit = 0:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits a Destination Unreachable Error Message for "Address Unreachable" to the RUT with the Hop Limit set to Zero in the IPv6 header and with a Destination Address set to an off-link address.

C. *Reception of Flawed Time Exceeded Code 0 with No Route To Destination:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits a Time Exceeded Error Message for "Hop Limit Exceeded in Transit" to the RUT with the Destination Address set to an off-link address that does not exist.

D. *Reception of Flawed Time Exceeded Code 1 with No Route To Destination:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits a Time Exceeded Error Message for "Fragment Reassembly Time Exceeded" to the RUT with the Destination Address set to an off-link address that does not exist.

E. *Reception of Flawed Packet Too Big with Address Unreachable:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits a Packet Too Big Error Message to the RUT with the Destination Address set to an on-link address that does not exist.

F. *Reception of Flawed Parameter Problem with Hop Limit = 0:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits a Parameter Problem Error Message to the RUT with the Hop Limit set to Zero in the IPv6 header and with a Destination Address set to an off-link address.

**Comments on Test Results**

A. The RUT did not send a Destination Unreachable Error Message with Code 3 to TN1 when it receives a Destination Unreachable Message with Code 0 for which it cannot resolve a destination address.

B. The RUT did not send a Time Exceeded message with Code 0 to TN1 when it receives a Destination Unreachable Message with Code 3 that contains a Hop Limit of 0.

C. The RUT did not send a Destination Unreachable Error Message with code 0 to TN1 when it receives a Time Exceeded Message with Code 0 for which it cannot route.

D. The RUT did not send a Destination Unreachable Error Message with code 0 to TN1 when it receives a Time Exceeded Message with Code 1 for which it cannot route.

E. The RUT did not send a Destination Unreachable Error Message with code 3 to TN1 when it receives a Packet Too Big Message for which it cannot resolve a destination address.

F. The RUT did not send a Time Exceeded Error Message with code 0 to TN1 when it receives a Parameter Problem Message that contains a Hop Limit of 0.

| Test # | | Result | |
|---|---|---|---|
| v6LC.5.1.10 | **Error Condition With Multicast Destination** | **A** | **PASS** |
| | | **B** | **PASS** |

**Purpose**: Verify that a node properly handles the reception of an error condition caused by a packet with a Multicast Destination Address.

**Comments on Test Procedure**

A. *UDP Port Unreachable:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits a UDP packet on Link B with the Destination Address set to the all-nodes link-local multicast address. The destination port is set to 9000. (Make sure the NUT is not listening on port 9000.)
B. *Echo Request Reassembly Timeout:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an ICMPv6 Echo Request Fragment to the all-nodes link-local multicast address. The offset of the fragment is 0 (the first fragment) and the More Fragments Flag is set.

**Comments on Test Results**

A. The NUT did not send a Destination Unreachable Error Message to TN1 when it receives a UDP packet for an unreachable port.
B. The NUT did not send a Time Exceeded Error Message to TN1 60 seconds after it receives the first fragment of an ICMPv6 Echo Request.

| Test # | | Result | |
|---|---|---|---|
| v6LC.5.1.11 | **Error Condition With Non-Unique Source - Unspecified** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |

**Purpose**: Verify that a node properly handles the reception of an error condition caused by a packet with a source address that does not uniquely identify a single node.

**Comments on Test Procedure**

A. *UDP Port Unreachable (Routers and Hosts):* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits a UDP Packet to the NUT's Global address with a Source Address set to the unspecified address (::). The destination port is set to 9000. (Make sure the NUT is not listening on port 9000.)

B. *Echo Request Too Big (Routers Only):* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. Configure the RUT with a link MTU equal to the IPv6 minimum link MTU (1280 octets) on its interface to Link A (to TN2) and Enable the RUT's interface to Link A. Configure all other interfaces on the RUT with the default link MTU for its associated media type. The link MTU for RUT's interface to Link A should be smaller than its link MTU to Link B. TN1 transmits an ICMPv6 Echo Request with a total message size of 1500 octets to TN2 with a first hop through the RUT. The Source Address is set to the unspecified address (::).

C. *Echo Request Reassembly Timeout (Routers and Hosts):* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an ICMPv6 Echo Request Fragment to the NUT. The offset of the fragment is 0 (the first fragment) and the More Fragments Flag is set. The Source Address is set to the unspecified address (::).

D. *Echo Request with Unknown Option in Destination Options (Routers and Hosts):* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an ICMPv6 Echo Request to the NUT. The Source Address is set to the unspecified address (::). It includes a Destination Options Header with the unrecognized Option of type 135. (Highest Order bits set to $10_b$).



**Comments on Test Results**

A. The NUT did not send a Destination Unreachable Error Message to TN1 when it receives a UDP packet for an unreachable port.

B. The RUT did not send a Packet Too Big Error Message to TN1 when it receives an ICMPv6 Echo Request that is too large for it to send on its outgoing interface.

C. The NUT did not send a Time Exceeded Error Message to TN1 60 seconds after it receives the first fragment of an ICMPv6 Echo Request.

D. The NUT did not send a Parameter Problem Error Message when it receives an ICMPv6 Echo Request with an unknown option with highest bits $10_b$.

| Test # | | Result | |
|---|---|---|---|
| v6LC.5.1.12 | **Error Condition With Non-Unique Source - Multicast** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |

**Purpose**: Verify that a node properly handles the reception of an error condition caused by a packet with a source address that does not uniquely identify a single node.

**Comments on Test Procedure**

A. *UDP Port Unreachable (Routers and Hosts):* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits a UDP Packet to the NUT's Global address with a Source Address set to the TN1's Solicited-Node Multicast address. The destination port is set to 9000. (Make sure the NUT is not listening on port 9000.)

B. *Echo Request Too Big (Routers Only):* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. Configure the RUT with a link MTU equal to the IPv6 minimum link MTU (1280 octets) on its interface to Link A (to TN2) and Enable the RUT's interface to Link A. Configure all other interfaces on the RUT with the default link MTU for its associated media type. The link MTU for RUT's interface to Link A should be smaller than its link MTU to Link B. TN1 transmits an ICMPv6 Echo Request with a total message size of 1500 octets to TN2 with a first hop through the RUT. The Source Address is set to TN1's Solicited-Node Multicast address.

C. *Echo Request Reassembly Timeout (Routers and Hosts):* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an ICMPv6 Echo Request Fragment to the NUT. The offset of the fragment is 0 (the first fragment) and the More Fragments Flag is set. The Source Address is set to the TN1's Solicited-Node Multicast address.

D. *Echo Request with Unknown Option in Destination Options (Routers and Hosts):* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an ICMPv6 Echo Request to the NUT. The Source Address is set to TN1's Solicited-Node Multicast address. It includes a Destination Options Header with the unrecognized Option of type 135. (Highest Order bits set to $10_b$).



**Comments on Test Results**

A. The NUT did not send a Destination Unreachable Error Message to TN1 when it receives a UDP packet for an unreachable port.

B. The RUT did not send a Packet Too Big Error Message to TN1 when it receives an ICMPv6 Echo Request that is too large for it to send on its outgoing interface.

C. The NUT did not send a Time Exceeded Error Message to TN1 60 seconds after it receives the first fragment of an ICMPv6 Echo Request.

D. The NUT did not send a Parameter Problem Error Message when it receives an ICMPv6 Echo Request with an unknown option with highest bits $10_b$.

| Test # | | Result | |
|---|---|---|---|
| v6LC.5.1.13 | **Error Condition With Non-Unique Source – Anycast (Routers Only)** | **A** | **PASS** |
| | | **B** | **PASS** |
| | | **C** | **PASS** |
| | | **D** | **PASS** |

**Purpose**: Verify that a node properly handles the reception of an error condition caused by a packet with a source address that does not uniquely identify a single node.

**Comments on Test Procedure**

A. *UDP Port Unreachable:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits a UDP Packet to the NUT's Global address with a Source Address set to the TR1's Subnet-Router Anycast Address. The destination port is set to 9000. (Make sure the NUT is not listening on port 9000.)

B. *Echo Request Too Big:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. Configure the RUT with a link MTU equal to the IPv6 minimum link MTU (1280 octets) on its interface to Link A (to TN2) and Enable the RUT's interface to Link A. Configure all other interfaces on the RUT with the default link MTU for its associated media type. The link MTU for RUT's interface to Link A should be smaller than its link MTU to Link B. TN1 transmits an ICMPv6 Echo Request with a total message size of 1500 octets to TN2 with a first hop through the RUT. The Source Address is set to TR1's Subnet-Router Anycast Address.

C. *Echo Request Reassembly Timeout:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an ICMPv6 Echo Request Fragment to the NUT. The offset of the fragment is 0 (the first fragment) and the More Fragments Flag is set. The Source Address is set to the TR1's Subnet-Router Anycast Address.

D. *Echo Request with Unknown Option in Destination Options:* TR1 transmits a Router Advertisement to the all-nodes multicast address. The Router Advertisement includes a Prefix Advertisement with a global prefix and the L and A bits set. TN1 transmits an ICMPv6 Echo Request to the NUT. The Source Address is set to TR1's Subnet-Router Anycast Address. It includes a Destination Options Header with the unrecognized Option of type 135. (Highest Order bits set to $10_b$).



**Comments on Test Results**

A. The NUT did not send a Destination Unreachable Error Message to TN1 when it receives a UDP packet for an unreachable port.

B. The RUT did not send a Packet Too Big Error Message to TN1 when it receives an ICMPv6 Echo Request that is too large for it to send on its outgoing interface.

C. The NUT did not send a Time Exceeded Error Message to TN1 60 seconds after it receives the first fragment of an ICMPv6 Echo Request.

D. The NUT did not send a Parameter Problem Error Message when it receives an ICMPv6 Echo Request with an unknown option with highest bits $10_b$.