IPv6 READY

Phase II Test Interoperability Specification
Core Protocols

Technical Document

Revision 2.7.4

IPv6 Forum *http://www.ipv6forum.org*
Converged Test Specification *http://www.ipv6ready.org*
TAHI Project (Japan)
UNH InterOperability Lab (USA)

MODIFICATION RECORD

Version 1.0 — September 2, 2004

Version 2.0 — September 8, 2004

Version 2.1 — September 10, 2004

Version 2.2 — September 12, 2004

Version 2.3 — September 27, 2004

Version 2.3.1 — October 3, 2004

Version 2.4 — December 9, 2004

- Test IP6Interop.1.4a, b

Changed Observable Results.
Step 9 and 15: TAR-Host1 MUST not transmit an Echo Reply using TAR-Router1 as its first hop.

Versions 2.5 — December 15, 2004

- Test IP6Interop.1.4a, b

Added Observable Results.
Step 9 and 15: TAR-Host1 MUST not transmit a multicast NS with a target set to TR1's link-local address.

Version 2.6 — January 18, 2005

- Test IP6Interop.1.2, Procedure
  Steps 1, 7, 13, 19, 25 and 31: changed "interface" to "device"
- Test IP6Interop.1.4b- Removed.
- Test IP6Interop.1.5, procedure /results: changed the direction of the ICMP Echo Request from REF-Host2 to TAR-Host1.
- Test IP6Interop.1.6b
  Added REF-Router1 to transmit RA with MTU=1280.

Version 2.6.1 — February 11, 2005

- Added Test Setup: to record info documentation.
- Added Phase I-II requirements to Tests performed on Host/Routers

Version 2.7.0 — March 29, 2005

- Test IP6Interop.1.2 all Parts, added step to check neighbor cache.
  - Parts A, C and E, specified boot-up order
  - Parts B, D and F, added DAD for both targets.
- Test IP6Interop.1.3, added initialization step to Test Setup.
- Test IP6Interop1.4, changed Step 5 to ping from REF-Host2
  - Fixed Observable results, Step 3 for on-link assumption.
- Test IP6Interop1.6 Part B, added new test: PMTU Discovery
- Test IP6Interop1.6 Part C, added steps 14 through 17.
- Test IP6Interop1.6 Part D and E, fixed results for steps 22 and 27.
- Test IP6Interop1.7 Part B, added steps 7 through 10.
- Removed references to Record interface information.
- Removed IP6Inerop1.6a from being a Phase-I requirement.

Version 2.7.1 — May 26, 2005

- Test IP6Interop.1.6 Part B, added steps 9-13, to reverse the roles of TAR-Routers.

Version 2.7.2 — June 2, 2005

- Test IP6Interop.1.6 part B, fixed topology
- Test IP6Interop.1.7 added requirement for Ref-Host

Version 2.7.3	June 7, 2005 • Fixed Typos.
Version 2.7.4	June 10, 2005 • Fixed Typos.

ACKNOWLEDGMENTS

The IPv6 Forum would like to acknowledge the efforts of the following organizations in the development of this test suite.

University of New Hampshire – InterOperability Laboratory
TAHI Project

INTRODUCTION

Overview

The IPv6 forum plays a major role to bring together industrial actors, to develop and deploy the new generation of IP protocols. Contrary to IPv4, which started with a small closed group of implementers, the universality of IPv6 leads to a huge number of implementations. Interoperability has always been considered as a critical feature in the Internet community.

Due to the large number of IPv6 implementations, it is important to provide the market a strong signal proving the level of interoperability across various products.

To avoid confusion in the mind of customers, a globally unique logo programme should be defined. The IPv6 logo will give confidence to users that IPv6 is currently operational. It will also be a clear indication that the technology will still be used in the future. To summarize, this logo programme will contribute to the feeling that IPv6 is available and ready to be used.

The IPv6 Logo Program consists of three phases:

Phase I
In a first stage, the Logo will indicate that the product includes IPv6 mandatory core protocols and can interoperate with other IPv6 implementations.

Phase II
The "IPv6 ready" step implies a proper care, technical consensus and clear technical references. The IPv6 ready logo will indicate that a product has successfully satisfied strong requirements stated by the IPv6 Logo Committee (v6LC).

To avoid confusion, the logo "IPv6 Ready" will be generic. The v6LC will define the test profiles with associated requirements for specific functionalities.

Phase III
Same as Phase 2 with IPsec mandated.

Abbreviations and Acronyms

DAD: Duplicate Address Detection
MTU: Maximum Transmission Unit
NCE: Neighbor Cache Entry
REF-Host: Reference Host
REF-Router: Reference Router
TAR-Host: Target Host
TAR-Router: Target Router
TLLA: Target Link-layer Address

TEST ORGANIZATION

This document organizes tests by Section based on related test methodology or goals. Each group begins with a brief set of comments pertaining to all tests within that group. This is followed by a series of description blocks; each block describes a single test. The format of the description block is as follows:

Test Label: The test label and title comprise the first line of the test block. The test label is composed by concatenating the short test suite name, the section number, the group number, and the test number within the group. These elements are separated by periods. The Test Number is the section, group and test number, also separated by periods.

Purpose: The Purpose is a short statement describing what the test attempts to achieve. It is usually phrased as a simple assertion of the feature or capability to be tested.

References: The References section lists cross-references to the specifications and documentation that might be helpful in understanding and evaluating the test and results.

Resource Requirements: The Resource Requirements section specifies the software, hardware, and test equipment that will be needed to perform the test.

Discussion: The Discussion is a general discussion of the test and relevant section of the specification, including any assumptions made in the design or implementation of the test as well as known limitations.

Test Setup: The Test Setup section describes the configuration of all devices prior to the start of the test. Different parts of the procedure may involve configuration steps that deviate from what is given in the test setup. If a value is not provided for a protocol parameter, then the protocol's default is used for that parameter.

Procedure: This section of the test description contains the step-by-step instructions for carrying out the test. These steps include such things as enabling interfaces, unplugging devices from the network, or sending packets from a test station. The test procedure also cues the tester to make observations, which are interpreted in accordance with the observable results given for that test part.

Observable Results: This section lists observable results that can be examined by the tester to verify that the RUT is operating properly. When multiple observable results are possible, this section provides a short discussion on how to interpret them. The determination of a pass or fail for each test is usually based on how the RUT's behavior compares to the results described in this section.

Possible Problems: This section contains a description of known issues with the test procedure, which may affect test results in certain situations.

REFERENCES

The following documents are referenced in this text:

[IPv6-SPEC] Hinden, R., S. Deering, Internet Protocol, Version 6 (IPv6) Specification, RFC 2460, December 1998.

[ND] Narten, T., Nordmark, E., and W. Simpson, Neighbor Discovery for IP Version 6 (IPv6), RFC 2461, December 1998.

[ADDRCONF] Thomson, S., T. Narten, IPv6 Stateless Address Autoconfiguration, RFC 2462, December 1998.

[ICMPv6] Conta, A., S. Deering, Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification, RFC 2463, December 1998.

[PMTU] McCann, J., S. Deering, and J. Mogul, Path MTU Discovery for IPv6, RFC 1981, August 1996.

[MLD] Deering, S., Fenner, W., Haberman, B., Multicast Listener Discovery (MLD) for IPv6, RFC 2710, October 1999.

[T/TCP] R. Braden, TCP Extensions for Transactions Functional Specification, RFC 1644, July 1994.

[FTP] J. Postel, J. Reynolds, File Transfer Protocol (FTP), RFC 959, October 1985.

[TELNET] J. Postel, J. Reynolds, TELNET Protocol Specification, RFC 854, May 1983.

[TFTP] K. Sollins, The TFTP Protocol (Revision 2), RFC 1350, July 1992.

[HTTP] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, Hypertext Transfer Protocol – HTTP/1.1, June 1999.

TABLE OF CONTENTS

Tests performed on Host/Router

The tests under the Host/Router column marked by an "X" must be performed for Phase-I or Phase-II as specified below. If there is no "X" listed under the Host/Router column, this test may be omitted for Phase-I and Phase-II.

	Hosts:	Routers:	Phase-I	Phase-II
IP6Interop.1.1a	X	-	X	X
IP6Interop.1.1b	X	-	X	X
IP6Interop.1.1c	X	-	-	X
IP6Interop.1.1d	X	X	X	X
IP6Interop.1.1e	X	X	X	X
IP6Interop.1.1f	X	X	-	X
IP6Interop.1.1g	-	X	X	X
IP6Interop.1.1h	-	X	X	X
IP6Interop.1.1i	-	X	-	X
IP6Interop.1.2a	X	-	-	X
IP6Interop.1.2b	X	-	-	X
IP6Interop.1.2c	X	X	-	X
IP6Interop.1.2d	X	X	-	X
IP6Interop.1.2e	-	X	-	X
IP6Interop.1.2f	-	X	-	X
IP6Interop.1.3a	X	X	-	X
IP6Interop.1.3b	X	X	-	X
IP6Interop.1.3c	X	X	-	X
IP6Interop.1.4	X	X	X	X
IP6Interop.1.5	X	X	-	X
IP6Interop.1.6a	X	X	-	X
IP6Interop.1.6b	X	X	-	X
IP6Interop.1.6c	X	-	-	X
IP6Interop.1.6d	X	X	-	X
IP6Interop.1.6e	-	X	-	X
IP6Interop.1.7a	X	X	-	X
IP6Interop.1.7b	-	X	-	X

Reference Requirements

Each reference node used in a test must be able to support the functionality required for that test.

Group 1: IPv6 Core Protocol and ICMPv6 Interoperability

Scope

Tests in this group verify that the target devices are able to engage in various aspects of the base IPv6 protocol.

Overview

The following tests verify operations such as ICMPv6, stateless address autoconfiguration, on-link determination, Duplicate Address Detection, path MTU discovery, fragmentation, redirects, communication when configured with multiple prefixes, and network renumbering

Test IP6Interop.1.1: ICMPv6 Echo Interoperability

Purpose: To verify that a successful ICMPv6 Echo Request, Echo Reply exchange can be achieved in two directions.

References:

- [ICMPv6] – Section 4

Resource Requirements:

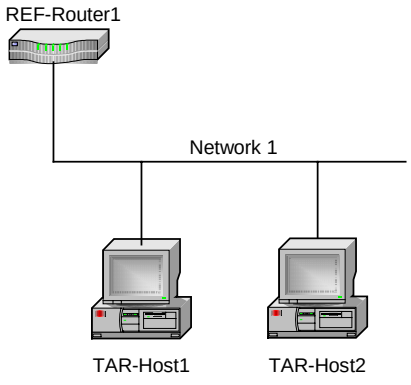
- Monitor to capture packets
- ping6 implementations

Discussion: Every node must implement an ICMPv6 Echo responder function that receives Echo Requests and sends corresponding Echo Replies. A node should also implement an application-layer interface for sending Echo Requests and receiving Echo Replies, for diagnostic purposes. The source address of an Echo Reply sent in response to a unicast Echo Request message must be the same as the destination address of that Echo Request message.

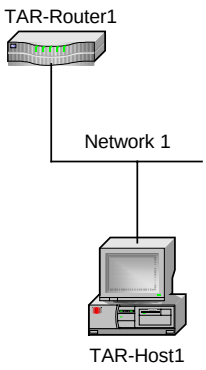
An Echo Reply SHOULD be sent in response to an Echo Request message sent to an IPv6 multicast address. The source address of the reply MUST be a unicast address belonging to the interface on which the multicast Echo Request message was received.

Test Setup: For each Part, connect hosts and routers as per the figures below. Allow time for all devices to perform stateless address autoconfiguration and Duplicate Address Detection and.

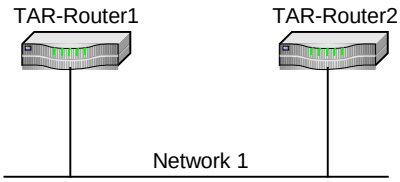
Parts A-C:



Parts D-F:



Parts G-I



Procedure:

Part A: Link-Local unicast address (Host vs Host)

1. Transmit ICMPv6 Echo Requests from TAR-Host1 to the link-local address of TAR-Host2.
2. Observe the packets on Network1.
3. Transmit ICMPv6 Echo Requests from TAR-Host2 to the link-local address of TAR-Host1.
4. Observe the packets on Network1.

Part B: Global Unicast Address (Host vs Host)

5. Transmit ICMPv6 Echo Requests from TAR-Host1 to the Global unicast address of TAR-Host2.
6. Observe the packets on Network1.
7. Transmit ICMPv6 Echo Requests from TAR-Host2 to the Global unicast address of TAR-Host1.
8. Observe the packets on Network1.

Part C: Multicast Address (Host vs Host)

9. Disable REF-Router1's interface on Network1.
10. Transmit ICMPv6 Echo Requests from TAR-Host1 to the All Nodes multicast address (FF02::1).
11. Observe the packets on Network1.
12. Transmit ICMPv6 Echo Requests from TAR-Host2 to the All Nodes multicast address (FF02::1).
13. Observe the packets on Network1.

Part D: Link-Local Unicast Address (Host vs Router)

14. Transmit ICMPv6 Echo Requests from TAR-Host1 to the link-local address of TAR-Router1.
15. Observe the packets on Network1.
16. Transmit ICMPv6 Echo Requests from TAR-Router1 to the link-local address of the TAR-Host1.
17. Observe the packets on Network1.

Part E: Global Unicast Address (Host vs Router)

18. Transmit ICMPv6 Echo Requests from TAR-Host1 to the Global unicast address of TAR-Router1.
19. Observe the packets on Network1.
20. Transmit ICMPv6 Echo Requests from TAR-Router1 to the Global unicast address of the TAR-Host1.
21. Observe the packets on Network1.

Part F: Multicast Address (Host vs Router)

22. Transmit ICMPv6 Echo Requests from TAR-Host1 to the All Nodes multicast address (FF02::1).
23. Observe the packets on Network1.

24. Transmit ICMPv6 Echo Requests from TAR-Router1 to the All Nodes multicast address (FF02::1).
25. Observe the packets on Network1.
26. Repeat Steps 22 through 23 transmitting an ICMPv6 Echo Request from TAR-Host1 to the All Routers multicast address (FF02::2).

Part G: Link-Local unicast address (Router vs Router)

27. Transmit ICMPv6 Echo Requests from TAR-Router1 to the link-local address of TAR-Router2.
28. Observe the packets on Network1.
29. Transmit ICMPv6 Echo Requests from TAR-Router2 to the link-local address of TAR-Router1.
30. Observe the packets on Network1.

Part H: Global Unicast Address (Router vs Router)

31. Transmit ICMPv6 Echo Requests from TAR-Router1 to the Global unicast address of TAR-Router2.
32. Observe the packets on Network1.
33. Transmit ICMPv6 Echo Requests from TAR-Router2 to the Global unicast address of TAR-Router1.
34. Observe the packets on Network1.

Part I: Multicast Address (Router vs Router)

35. Transmit ICMPv6 Echo Requests from TAR-Router1 to the All Nodes multicast address (FF02::1).
36. Observe the packets on Network1.
37. Transmit ICMPv6 Echo Requests from TAR-Router2 to the All Nodes multicast address (FF02::1).
38. Observe the packets on Network1.
39. Repeat Steps 35 through 38 transmitting an ICMPv6 Echo Request to the All Routers multicast address (FF02::2).

Observable Results:

- *Parts A and B*

  Step 2, 6: TAR-Host2 must receive all the ICMPv6 Echo Requests sent from TAR-Host1 and respond with ICMPv6 Echo Replies. The Source Address of the Echo Reply must be equal to the Destination Address that was in the Echo Request, and the Destination Address of the Echo Reply must be equal to the Source Address that was in the Echo Request.

  Step 4, 8: TAR-Host1 must receive all the ICMPv6 Echo Requests sent from TAR-Host2 and respond with ICMPv6 Echo Replies. The Source Address of the Echo Reply must be equal to the Destination Address that was in the Echo Request, and the Destination Address of the Echo Reply must be equal to the Source Address that was in the Echo Request.

- *Part C*

  Step 11: TAR-Host2 should receive all the ICMPv6 Echo Requests sent from TAR-Host1 and respond with ICMPv6 Echo Replies. The Source Address of the Echo Reply must be equal to TAR-Host2's address, and the Destination Address of the Echo Reply must be equal to the Source Address that was in the Echo Request.

  Step 13: TAR-Host1 should receive all the ICMPv6 Echo Requests sent from TAR-Host2 and respond with ICMPv6 Echo Replies. The Source Address of the Echo Reply must be

equal to TAR-Host1's address, and the Destination Address of the Echo Reply must be equal to the Source Address that was in the Echo Request.

- *Parts D and E*

  Step 15, 19: TAR-Router1 must receive all the ICMPv6 Echo Requests sent from TAR-Host1 and respond with ICMPv6 Echo Replies. The Source Address of the Echo Reply must be equal to the Destination Address that was in the Echo Request, and the Destination Address of the Echo Reply must be equal to the Source Address that was in the Echo Request.

  Step 17, 21: TAR-Host1 must receive all the ICMPv6 Echo Requests sent from TAR-Router1 and respond with ICMPv6 Echo Replies. The Source Address of the Echo Reply must be equal to the Destination Address that was in the Echo Request, and the Destination Address of the Echo Reply must be equal to the Source Address that was in the Echo Request.

- *Part F*

  Step 23, 26: TAR-Router1 should receive all the ICMPv6 Echo Requests sent from TAR-Host1 and respond with ICMPv6 Echo Replies. The Source Address of the Echo Reply must be equal to TAR-Router1's address, and the Destination Address of the Echo Reply must be equal to the Source Address that was in the Echo Request.

  Step 25: TAR-Host1 should receive all the ICMPv6 Echo Requests sent from TAR-Router1 and respond with ICMPv6 Echo Replies. The Source Address of the Echo Reply must be equal to TAR-Host1's address, and the Destination Address of the Echo Reply must be equal to the Source Address that was in the Echo Request.

- *Parts G and H*

  Step 28, 32: TAR-Router2 must receive all the ICMPv6 Echo Requests sent from TAR-Router1 and respond with ICMPv6 Echo Replies. The Source Address of the Echo Reply must be equal to the Destination Address that was in the Echo Request, and the Destination Address of the Echo Reply must be equal to the Source Address that was in the Echo Request.

  Step 30, 34: TAR-Router1 must receive all the ICMPv6 Echo Requests sent from TAR-Router2 and respond with ICMPv6 Echo Replies. The Source Address of the Echo Reply must be equal to the Destination Address that was in the Echo Request, and the Destination Address of the Echo Reply must be equal to the Source Address that was in the Echo Request.

- *Part I*

  Step 36: TAR-Router2 should receive all the ICMPv6 Echo Requests sent from TAR-Router1 and respond with ICMPv6 Echo Replies. The Source Address of the Echo Reply must be equal to TAR-Router2's address, and the Destination Address of the Echo Reply must be equal to the Source Address that was in the Echo Request.

  Step 38: TAR-Router1 should receive all the ICMPv6 Echo Requests sent from TAR-Router2 and respond with ICMPv6 Echo Replies. The Source Address of the Echo Reply must be equal to TAR-Router1's address, and the Destination Address of the Echo Reply must be equal to the Source Address that was in the Echo Request.

Possible Problems:

- A passive node may not implement an application for sending Echo Requests.

Test IP6Interop.1.2: Address Autoconfiguration and Duplicate Address Detection

Purpose: To verify that a device can properly initialize on a network and communicate with other on-link partners.

References:

- [ADDRCONF] – Section 1, 5.4

Resource Requirements:

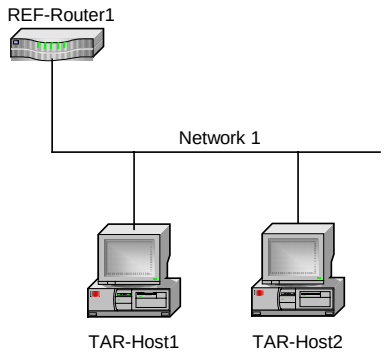
- Monitor to capture packets
- ping6 implementations

Discussion: When a host initializes on a given link, it performs stateless address autoconfiguration and Duplicate Address Detection. To insure that all configured addresses are likely to be unique on a given link, hosts run the Duplicate Address Detection algorithm on addresses before assigning them to an interface. The Duplicate Address Detection algorithm is performed on all addresses, independent of whether they are obtained via stateless or stateful autoconfiguration. In addition, routers perform Duplicate Address Detection on all addresses prior to assigning them to an interface.

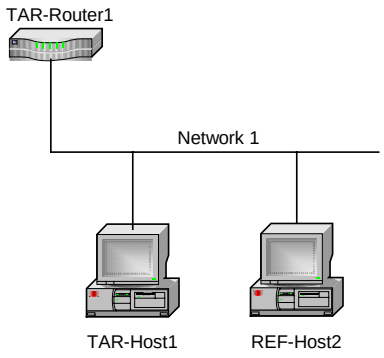
A tentative address that is determined to be a duplicate as described above, MUST NOT be assigned to an interface and the node SHOULD log a system management error. If the address is a link-local address formed from an interface identifier, the interface SHOULD be disabled.

Test Setup: Connect all devices as per the figure below.

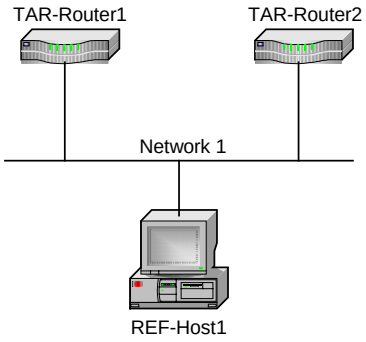
For Parts A-B



For Parts C-D:



For Parts E-F:



Procedure:

Part A: Duplicate Address Detection- Tentative Address Unique (Host vs Host)

1. Disable all interfaces connected to Network1.
2. Enable all interfaces on Network1, enabling TAR-Host1 before TAR-Host2.
3. Allow time for all devices on Network1 to perform Stateless Address Autoconfiguration and Duplicate Address Detection. Enable REF-Router1 making sure it has cleared its neighbor cache.
4. Transmit ICMPv6 Echo Request from REF-Router1 to the Link-Local Address of TAR-Host1.
5. Transmit ICMPv6 Echo Request from REF-Router1 to the Link-Local Address of TAR-Host2.
6. Observe the packets transmitted on Network1.
7. Repeat Steps 1 through 6, but in Step 2 enable TAR-Host2 before TAR-Host1.

Part B: Duplicate Address Detection- Tentative Address Duplicated (Host vs Host)

8. Disable all interfaces connected to Network1.
9. Configure TAR-Host2 to have the same Link-local Address as TAR-Host1.
10. Enable all interfaces on Network1, enabling TAR-Host2 before TAR-Host1.
11. Allow time for all devices to perform Stateless Address Autoconfiguration and Duplicate Address Detection. Enable REF-Router1 making sure it has cleared its neighbor cache.
12. Transmit ICMPv6 Echo Request from REF-Router1 to the Link-Local Address of TAR-Host1.
13. Observe the packets transmitted on Network1.
14. Disable all interfaces connected to Network1 and un-do the configured address on TAR-Host2.
15. Configure TAR-Host1 to have the same Link-local Address as TAR-Host2.
16. Enable all interfaces on Network1, enabling TAR-Host1 before TAR-Host2.
17. Allow time for all devices to perform Stateless Address Autoconfiguration and Duplicate Address Detection. Enable REF-Router1 making sure it has cleared its neighbor cache.
18. Transmit ICMPv6 Echo Request from REF-Router1 to the Link-Local Address of TAR-Host2.
19. Observe the packets transmitted on Network1.

Part C: Duplicate Address Detection- Tentative Address Unique (Host vs Router)

20. Disable all interfaces connected to Network1.
21. Enable all interfaces on Network1, enabling TAR-Host1 before TAR-Router1.
22. Allow time for all devices on Network1 to perform Stateless Address Autoconfiguration and Duplicate Address Detection. Enable REF-Host2 making sure it has cleared its neighbor cache.
23. Transmit ICMPv6 Echo Request from REF-Host2 to the Link-Local Address of TAR-Host1.
24. Transmit ICMPv6 Echo Request from REF-Host2 to the Link-Local Address of TAR-Router1.
25. Observe the packets transmitted on Network1.
26. Repeat Steps 20 through 25, but in Step 21 enable TAR-Router1 before TAR-Host1.

Part D: Duplicate Address Detection- Tentative Address Duplicated (Host vs Router)

27. Disable all interfaces connected to Network1.
28. Configure TAR-Router1 to have the same Link-local Address as TAR-Host1.
29. Enable all interfaces, enabling TAR-Router1 before TAR-Host1.
30. Allow time for all devices to perform Stateless Address Autoconfiguration and Duplicate Address Detection. Enable REF-Host2 making sure it has cleared its neighbor cache.
31. Transmit ICMPv6 Echo Request from REF-Host2 to the Link-Local Address of TAR-Host1.
32. Observe the packets transmitted on Network1.
33. Disable all interfaces connected to Network1 and un-do the configured address on TAR-Router1.
34. Configure TAR-Host1 to have the same Link-local Address as TAR-Router1.
35. Enable all interfaces on Network1, enabling TAR-Host1 before TAR-Router1.
36. Allow time for all devices to perform Stateless Address Autoconfiguration and Duplicate Address Detection. Enable REF-Host2 making sure it has cleared its neighbor cache.
37. Transmit ICMPv6 Echo Request from REF-Host2 to the Link-Local Address of TAR-Router1.
38. Observe the packets transmitted on Network1.

Part E: Duplicate Address Detection- Tentative Address Unique (Router vs Router)

39. Disable all interfaces connected to Network1.
40. Enable all interfaces on Network1, enabling TAR-Router1 before TAR-Router2.
41. Allow time for all devices on Network1 to perform Stateless Address Autoconfiguration and Duplicate Address Detection. Enable REF-Host1 making sure it has cleared its neighbor cache.
42. Transmit ICMPv6 Echo Request from REF-Host1 to the Link-Local Address of TAR-Router1.
43. Transmit ICMPv6 Echo Request from REF-Host1 to the Link-Local Address of TAR-Router2.
44. Observe the packets transmitted on Network1.
45. Repeat Steps 39 through 44, but in Step 40 enable TAR-Router2 before TAR-Router1.

Part F: Duplicate Address Detection- Tentative Address Duplicated (Router vs Router)

46. Disable all interfaces connected to Network1.
47. Configure TAR-Router2 to have the same Link-local Address as TAR-Router1.
48. Enable all interfaces on Network1, enabling TAR-Router2 before TAR-Router1.
49. Allow time for all devices to perform Stateless Address Autoconfiguration and Duplicate Address Detection. Enable REF-Host1 making sure it has cleared its neighbor cache.
50. Transmit ICMPv6 Echo Request from REF-Host1 to the Link-Local Address of TAR-Router1.
51. Observe the packets transmitted on Network1.
52. Disable all interfaces connected to Network1 and un-do the configured address on TAR-Router2.
53. Configure TAR-Router1 to have the same Link-local Address as TAR-Router2.
54. Enable all interfaces on Network1, enabling TAR-Router1 before TAR-Router2.
55. Allow time for all devices to perform Stateless Address Autoconfiguration and Duplicate Address Detection. Enable REF-Host1 making sure it has cleared its neighbor cache.
56. Transmit ICMPv6 Echo Request from REF-Host1 to the Link-Local Address of TAR-Router2.
57. Observe the packets transmitted on Network1.

Observable Results:

- *Part A*

  Step 6: TAR-Host1 and TAR-Host2 must respond to any Neighbor Solicitations transmitted by REF-Router1 by transmitting a Neighbor Advertisement. TAR-Host1 and TAR-Host2 must respond to the ICMPv6 Echo Requests transmitted by REF-Router1.
- *Part B*

Step 13: TAR-Host2 must respond to any Neighbor Solicitations transmitted by REF-Router1 by transmitting a Neighbor Advertisement. TAR-Host2, and not TAR-Host1, must respond to the ICMPv6 Echo Requests transmitted by REF-Router1.
Step 19: TAR-Host1 must respond to any Neighbor Solicitations transmitted by REF-Router1 by transmitting a Neighbor Advertisement. TAR-Host1, and not TAR-Host2, must respond to the ICMPv6 Echo Requests transmitted by REF-Router1.

- *Part C*

  Step 25: TAR-Host1 and TAR-Router1 must respond to any Neighbor Solicitations transmitted by REF-Host2 by transmitting a Neighbor Advertisement. TAR-Host1 and TAR-Router1 must respond to the ICMPv6 Echo Requests transmitted by REF-Host2.
- *Part D*

  Step 32: TAR-Router1 must respond to any Neighbor Solicitations transmitted by REF-Host2 by transmitting a Neighbor Advertisement. TAR-Router1, and not TAR-Host1, must respond to the ICMPv6 Echo Requests transmitted by REF-Host2.
  Step 38: TAR-Host1 must respond to any Neighbor Solicitations transmitted by REF-Host2 by transmitting a Neighbor Advertisement. TAR-Host1, and not TAR-Router1, must respond to the ICMPv6 Echo Requests transmitted by REF-Host2.
- *Part E*

  Step 44: TAR-Router1 and TAR-Router2 must respond to any Neighbor Solicitations transmitted by REF-Host1 by transmitting a Neighbor Advertisement. TAR-Router1 and TAR-Router2 must respond to the ICMPv6 Echo Requests transmitted by REF-Host1.
- *Part F*

  Step 51: TAR-Router2 must respond to any Neighbor Solicitations transmitted by REF-Host1 by transmitting a Neighbor Advertisement.TAR-Router2, and not TAR-Router1, must respond to the ICMPv6 Echo Requests transmitted by REF-Host1.
  Step 57: TAR-Router1 must respond to any Neighbor Solicitations transmitted by REF-Host1 by transmitting a Neighbor Advertisement. TAR-Router1, and not TAR-Router2, must respond to the ICMPv6 Echo Requests transmitted by REF-Host1.

Possible Problems:

- A host or router may not bind the new link-local address if it has its pre-configured link-local address to fall back on. For this reason, the original link-local address should be removed from the testing interface. If the original cannot be removed then the global addresses may be used.

Test IP6Interop.1.3: Processing Router Advertisments- Prefix Discovery

Purpose: To verify that a device can properly perform prefix discovery.

References:

- [ND] – Sections 6.3.4, 6.3.5, and 12
- [ADDRCONF] – Section 4.1

Resource Requirements:

- Monitor to capture packets
- ping6 implementations

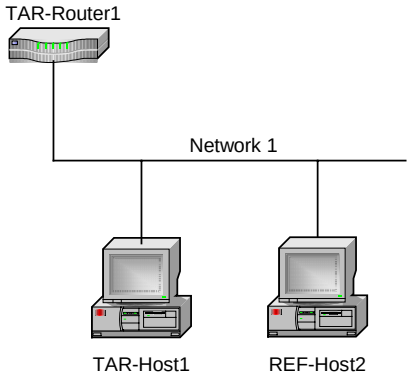
Discussion: For each Prefix Information option with the on-link flag set, a host does the following:
- If the prefix is the link-local prefix, silently ignore the Prefix Information option.
- If the prefix is not already present in the Prefix List, and the Prefix Information option's Valid Lifetime field is non-zero, create a new entry for the prefix and initialize its invalidation timer to the Valid Lifetime value in the Prefix Information option.
- If the prefix is already present in the host's Prefix List as the result of a previously-received advertisement, reset its invalidation timer to the Valid Lifetime value in the Prefix Information option. If the new Lifetime value is zero, time-out the prefix immediately (see Section 6.3.5).
- If the Prefix Information option's Valid Lifetime field is zero, and the prefix is not present in the host's Prefix List, silently ignore the option.

Whenever the invalidation timer expires for a Prefix List entry, that entry is discarded.

Multiple prefixes can be associated with the same link. By default, hosts learn all on-link prefixes from Router Advertisements. A host with multiple on-link prefixes should be able to communicate using any configured prefix, as long as its lifetime is still valid. Further, the Neighbor Discovery protocol together with IPv6 Address Autoconfiguration provides mechanisms to aid in renumbering - new prefixes and addresses can be introduced and old ones can be deprecated and removed.

In addition to sending periodic, unsolicited advertisements, a router sends advertisements in response to valid solicitations received on an advertising interface. A router MAY choose to unicast the response directly to the soliciting host's address (if the solicitation's source address is not the unspecified address), but the usual case is to multicast the response to the all-nodes group. In the latter case, the interface's interval timer is reset to a new random value, as if an unsolicited advertisement had just been sent (see Section 6.2.4). Router Advertisements sent in response to a Router Solicitation MUST be delayed by a random time between 0 and MAX_RA_DELAY_TIME seconds.

Test Setup: Connect hosts TAR-Host1 and REF-Host2 and router TAR-Router1 to Network1, per the figure below. Re-initialize each interface on Network 1 before each part.



Procedure:

Part A: Single Prefix Discovery (Host vs Router)

1. Configure TAR-Router1 to transmit Router Advertisements with one Prefix (valid lifetime > 0).
2. Administratively bring down the interface on TAR-Host1 that is connected to Network1.
3. Remove all Global Addresses from the interface on TAR-Host1 that is connected to Network1.
4. Administratively bring up the interface on TAR-Host1 that is connected to Network1 and allow time for TAR-Host1 and REF-Host2 to perform stateless address autoconfiguration and Duplicate Address Detection.
5. Transmit an ICMPv6 Echo Request from REF-Host2 to the Global Address of the TAR-Host1.
6. Observe the packets transmitted on Network1.

Part B: Multiple Prefix Discovery (Host vs Router)

7. Configure TAR-Router1 to transmit Router Advertisements with two prefixes: Prefix1, Prefix2 (valid lifetimes > 0) and allow time for TAR-Host1 and REF-Host2 to perform stateless address autoconfiguration and Duplicate Address Detection.
8. Transmit ICMPv6 Echo Requests from REF-Host2 to the Global Address of TAR-Host1 associated with Prefix1.
9. Observe the packets transmitted on Network1.
10. Transmit an ICMPv6 Echo Request from REF-Host2 to the Global Address of the TAR-Host1 associated with Prefix2.
11. Observe the packets transmitted on Network1.

Part C: Prefix Lifetime expires (Host vs Router)

12. Configure TAR-Router1 to transmit Router Advertisements with Prefix1 (valid lifetime = 30sec) and allow time for TAR-Host1 and REF-Host2 to perform stateless address autoconfiguration and Duplicate Address Detection.
13. Transmit an ICMPv6 Echo Request from REF-Host2 to the Global Address of the TAR-Host1 associated with Prefix1.
14. Observe the packets transmitted on Network1.
15. Allow 35 seconds to pass.
16. Transmit an ICMPv6 Echo Request from REF-Host2 to the Global Address of the TAR-Host1 associated with Prefix1.
17. Observe the packets transmitted on Network1.

Observable Results:

- *Part A*

  Step 6: TAR-Host1 must respond to all ICMPv6 Echo Requests from REF-Host2 with ICMPv6 Echo Replies.
- *Part B*

  Step 9: TAR-Host1 must respond to ICMPv6 Echo Requests from REF-Host2 with ICMPv6 Echo Replies associated with Prefix1.

  Step 11: TAR-Host1 must respond to ICMPv6 Echo Requests from REF-Host2 with ICMPv6 Echo Replies associated with Prefix2.
- *Part C*

  Step 14: TAR-Host1 must respond to ICMPv6 Echo Requests from REF-Host2 with ICMPv6 Echo Replies associated with Prefix1.

  Step 17: TAR-Host1 must timeout its Prefix1. TAR-Host1 must not respond to ICMPv6 Echo Requests from the TAR-HOST1 with ICMPv6 Echo Replies associated with Prefix1.

Possible Problems:

- None.

Test IP6Interop.1.4: Processing Router Advertisements- Router Lifetime (Host vs Router)

Purpose: To verify that a device can properly perform Router Discovery.

References:

- [ND] – Section 6.3.4

Resource Requirements:

- Monitor to capture packets
- ping6 implementations

Discussion: On receipt of a valid Router Advertisement, a host extracts the source address of the packet and does the following:

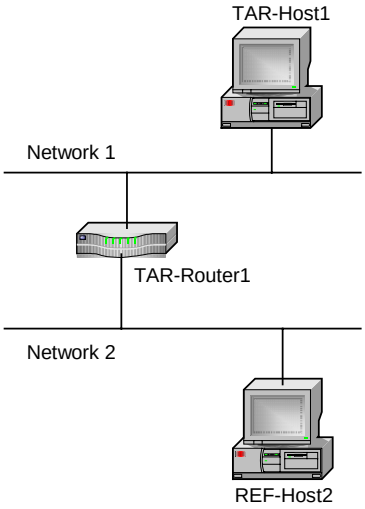
- If the address is not already present in the host's Default Router List, and the advertisement's Router Lifetime is non-zero, create a new entry in the list, and initialize its invalidation timer value from the advertisement's Router Lifetime field.

- If the address is already present in the host's Default Router List as a result of a previously-received advertisement, reset its invalidation timer to the Router Lifetime value in the newly-received advertisement.

- If the address is already present in the host's Default Router List and the received Router Lifetime value is zero, immediately time-out the entry as specified in Section 6.3.5.

To limit the storage needed for the Default Router List, a host MAY choose not to store all of the router addresses discovered via advertisements. However, a host MUST retain at least two router addresses and SHOULD retain more. Default router selections are made whenever communication to a destination appears to be failing. Thus, the more routers on the list, the more likely an alternative working router can be found quickly (e.g., without having to wait for the next advertisement to arrive).

Test Setup: Connect hosts TAR-Host1 and REF-Host2 and router TAR-Router1 to Network1 and Network2, per the figure below. Router TAR-Router1 route between Network1 and Network2. Allow time for the TAR-Host1 and REF-Host2 to perform stateless address autoconfiguration and Duplicate Address Detection.



Procedure:

1. Configure TAR-Router1 to transmit Router Advertisements with Router Lifetimes equal to 0 and at a normal interval on Network1, and Router Lifetimes greater than the Router Advertisement Interval on Network2.
2. Transmit an ICMPv6 Echo Request from REF-Host2 to the Global Address of TAR-Host1.
3. Observe the packets sent on Network1 and Network2.
4. Configure TAR-ROUTER1 to transmit Router Advertisements with Router Lifetimes set to 600 seconds and Router Advertisement Intervals set to approximately 60 seconds on both Network1 and Network2.
5. Transmit an ICMPv6 Echo Request from REF-Host2 to the Global Address of TAR-Host1.
6. Observe the packets sent on Network1 and Network2.
7. Configure TAR-Router1 to transmit Router Advertisements with the Router Lifetime set to 0 on Network1.
8. Transmit an ICMPv6 Echo Request from REF-Host2 to the Global Address of TAR-Host1.
9. Observe the packets transmitted on Network1 and Network2.

Observable Results:

Step 3: TAR-Host1 MUST not transmit an Echo Reply using TAR-Router1 as its first hop or transmit a multicast NS with a target address set to TR1's link-local address.
Step 6: REF-Host2 should send an Echo Request with a Destination Address of the TAR-Host1's Global Address. TAR-Host1 must use TAR-Router1 as its first hop for Network2 and the Echo Reply should be visible on Network2.
Step 9: TAR-Host1 MUST not transmit an Echo Reply using TAR-Router1 as its first hop or transmit a multicast NS with a target address set to TR1's link-local address.

Possible Problems:

- None.

Test IP6Interop.1.5: Redirect Function (Host vs Router)

Purpose: Verify the correct interoperability between a device's redirect handling with that of various IPv6 implementations.

References:

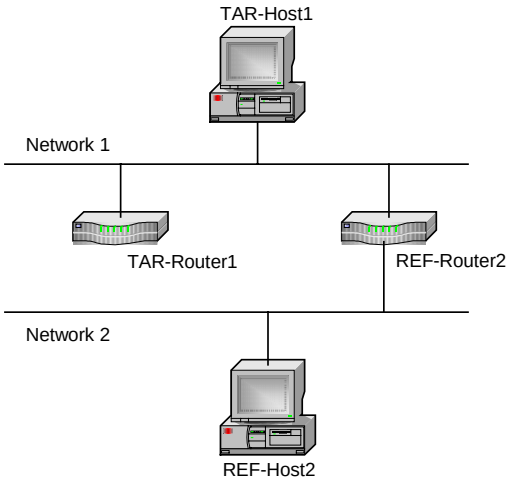
- [ND] – Section 4.5, 4.6.3, 8

Resource Requirements:

- Monitor to capture packets
- ping6 implementations

Discussion: Routers send redirect packets to inform a host of a better first-hop node on the path to a destination. Hosts can be redirected to a better first-hop router, but can also be informed by a redirect that the destination is in fact a neighbor. The latter is accomplished by having the ICMPv6 Target Address be equal to the ICMPv6 Destination Address in the redirect message.

Test Setup: Connect all devices, per the figure below. Configure router REF-Router2 to NOT transmit Router Advertisements on Network1. Router TAR-Router1 is not connected to Network2. Configure a static route on TAR-Router1 indicating REF-Router2's Link-local address as the next hop for network Network2. Router REF-Router2 routes between Network1 and Network2. Allow time for the TAR-Host1 and REF-Host2 to perform stateless address autoconfiguration and Duplicate Address Detection.



Procedure:

1. Transmit an ICMPv6 Echo Request from REF-Host2 to the Global Address of TAR-Host1.
2. Allow time for TAR-Router1 to send an ICMPv6 Redirect message to TAR-Host1 specifying REF-Router2 as a better first hop.
3. Observe the packets transmitted on Network1.

4. Transmit an ICMPv6 Echo Request from REF-Host2 to the Global Address of TAR-Host1.
5. Observe the packets transmitted on Network1 and Network2.
6. Remove the static route on TAR-Router1 configured in the test setup.

Observable Results:

Step 3: TAR-Router1 should send an ICMPv6 Redirect message to the TAR-Host1 indicating REF-Router2 as a better first hop to Network2.
Step 5: TAR-Host1 must use REF-Router2 as its first hop for the Echo Reply destined for Network2, indicating that it processed the ICMPv6 Redirect Message and applied it to its Routing Table. REF-Host2 must receive the Echo Reply from TAR-Host1.

Possible Problems:

- None.

Test IP6Interop.1.6: Path MTU Discovery and Fragmentation

Purpose: Verify that devices can participate in path MTU discovery and handle fragmentation in an IPv6 network.

References:

- [PMTU] – Section 3,4
- [ICMPv6] – Section 3.2

Resource Requirements:

- Monitor to capture packets
- ping6 implementations capable of sending large packets

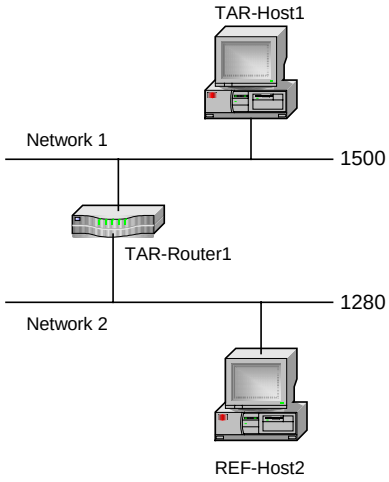
Discussion: IPv6 nodes should implement Path MTU Discovery in order to discover and take advantage of paths with PMTU greater than the IPv6 minimum link MTU. A source node initially assumes that the PMTU of a path is the (known) MTU of the first hop in the path. If any of the packets sent on that path are too large to be forwarded by some node along the path, that node will discard them and return ICMPv6 Packet Too Big messages. Upon receipt of such a message, the source node reduces its assumed PMTU for the path based on the MTU of the constricting hop as reported in the Packet Too Big message.

A Packet Too Big MUST be sent by a router in response to a packet that it cannot forward because the packet is larger than the MTU of the outgoing link. The information in this message is used as part of the Path MTU Discovery process [PMTU].

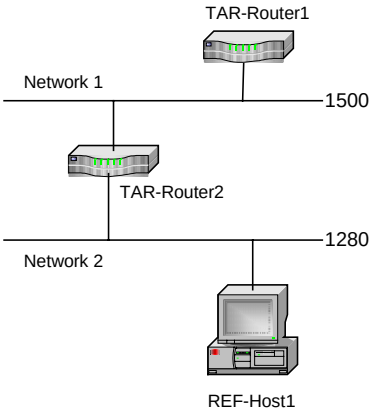
Nodes MAY detect increases in PMTU, but because doing so requires sending packets larger than the current estimated PMTU, and because the likelihood is that the PMTU will not have increased, this MUST be done at infrequent intervals. An attempt to detect an increase (by sending a packet larger than the current estimate) MUST NOT be done less than 5 minutes after a Packet Too Big message has been received for the given path. The recommended setting for this timer is twice its minimum value (10 minutes).

Test Setup: For Part A, connect hosts TAR-HOST1 and REF-HOST1 and router TAR-ROUTER1 to Network1 and Network2, per the figure below. Router TAR-ROUTER1 routes between Network1 and Network2. Allow time for the TAR-HOST1 and REF-HOST1 to perform stateless address autoconfiguration and Duplicate Address Detection. For Parts B through D, connect all devices as per the figure below. Re-boot each device after each part is performed.

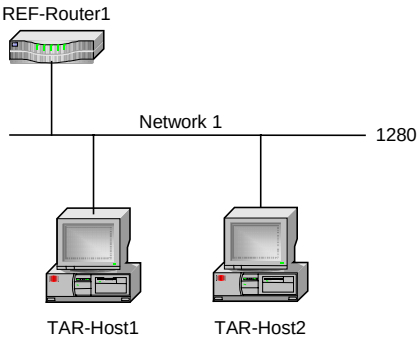
Part A:



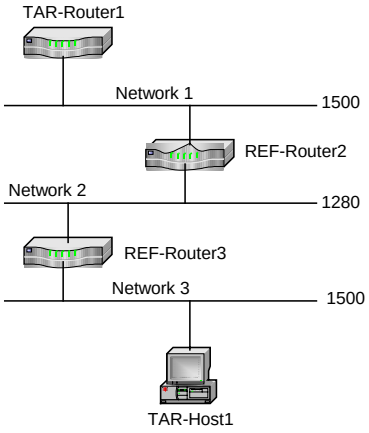
Part B:



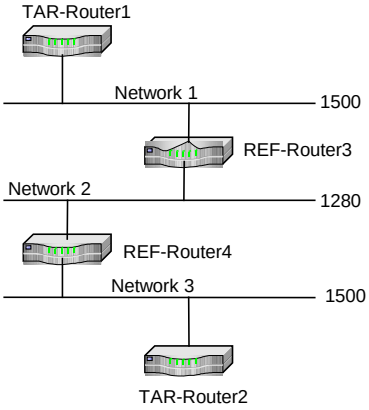
Part C:



Part D:



Part E:



Procedure:

Part A: PMTU Discovery (Host vs Router)

1. Configure the Network1 interface on TAR-Router1 with a path MTU of 1500 bytes.
2. Configure the Network2 interface on TAR-Router1 with a path MTU of 1280 bytes.
3. Transmit 1400 byte ICMPv6 Echo Requests from the Global Address of REF-Host2 to the Global Address of TAR-Host1.
4. Observe the packets transmitted on Network1 and Network2.

Part B: PMTU Discovery (Router vs Router)

5. Configure the Network1 interface on TAR-Router2 and TAR-Router1 with a path MTU of 1500 bytes.
6. Configure the Network2 interface on TAR-Router2 with a path MTU of 1280 bytes.
7. Transmit 1400 byte ICMPv6 Echo Requests from the Global Address of REF-Host1 to the Global Address of TAR-Router1.
8. Observe the packets transmitted on Network1 and Network2.
9. Disconnect TAR-Router1 and TAR-Router2 and switch the roles such that TAR-Router1 is routing between Network1 and Network2 and TAR-Router2 is connected to Network1 only.
10. Configure the Network1 interface on TAR-Router1 and TAR-Router2 with a path MTU of 1500 bytes.
11. Configure the Network2 interface on TAR-Router1 with a path MTU of 1280 bytes.
12. Transmit 1400 byte ICMPv6 Echo Requests from the Global Address of REF-Host1 to the Global Address of TAR-Router2.
13. Observe the packets transmitted on Network1 and Network2.

Part C: Fragmentation/Reassembly (Host vs Host)

14. Configure REF-Router1 to transmit a Router Advertisement with MTU=1280 on Network 1.
15. Transmit 1400 byte ICMPv6 Echo Requests from the link-local address of TAR-Host1 to the link-local address of TAR-Host2.
16. Observe the packets transmitted on Network1.
17. Transmit 1400 byte ICMPv6 Echo Requests from the link-local address of TAR-Host2 to the link-local address of TAR-Host1.
18. Observe the packets transmitted on Network1.
19. Transmit 1400 byte ICMPv6 Echo Requests from the on-link global address of TAR-Host1 to the on-link global address of TAR-Host2.
20. Observe the packets transmitted on Network1.
21. Transmit 1400 byte ICMPv6 Echo Requests from the on-link global address of TAR-Host2 to the on-link global address of TAR-Host1.
22. Observe the packets transmitted on Network1.

Part D: Fragmentation/Reassembly (Host vs Router)

23. Configure a static route on TAR-Router1 for Network 3 using a nexthop of REF-Router2. Configure static routes for Networks 1 and 3 on REF-Router1 and REF-Router3 so they are each the appropriate nexthop. Configure REF-Router3 to transmit Router Advertisements with a Router Lifetime > 0 on Network 3.
24. Transmit 1400 byte ICMPv6 Echo Requests from TAR-Router1 to TAR-Host1.
25. Observe the packets transmitted on Network1.

26. Transmit 1400 byte ICMPv6 Echo Requests from TAR-Host1 to TAR-Router1.
27. Observe the packets transmitted on Network1.

Part E: Fragmentation/Reassembly (Router vs Router)

28. Configure a static route on TAR-Router1 for Network 3 using a nexthop of REF-Router3. Configure static routes for Networks 1 and 3 on REF-Router3 and REF-Router4 so they are each the appropriate nexthop. Configure a static route on TAR-Router2 for Network 1 using a nexthop of REF-Router4.
29. Transmit 1400 byte ICMPv6 Echo Requests from TAR-Router1 to TAR-Router2.
30. Observe the packets transmitted on Network1.
31. Transmit 1400 byte ICMPv6 Echo Requests from TAR-Router2 to TAR-Router1.
32. Observe the packets transmitted on Network1.

Observable Results:

- *Part A*

  Step 4: TAR-Host1 should attempt to send the Echo Reply without fragmenting. TAR-Router1 must send an ICMPv6 Packet Too Big Message, and the TAR-Host1 must lower its path MTU estimate and fragment the Echo Reply.
- *Part B*

  Step 8: TAR-Router1 should attempt to send the Echo Reply without fragmenting. TAR-Router2 must send an ICMPv6 Packet Too Big Message, and the TAR-Router1 must lower its path MTU estimate and fragment the Echo Reply.

  Step 13: TAR-Router2 should attempt to send the Echo Reply without fragmenting. TAR-Router1 must send an ICMPv6 Packet Too Big Message, and the TAR-Router2 must lower its path MTU estimate and fragment the Echo Reply.
- *Part C*

  Step 16 and 20: TAR-Host1 must send the Echo Request with fragmenting. TAR-Host2 must respond to the fragmented Echo Request sent by TAR-Host1 with an Echo Reply.

  Step 18 and 22: TAR-Host2 must send the Echo Request with fragmenting. TAR-Host1 must reassemble and respond to the fragmented Echo Request with an Echo Reply.
- *Part D*

  Step 25: TAR-Router1 should attempt to send the Echo Request without fragmenting. REF-Router1 must send an ICMPv6 Packet Too Big Message, and the TAR-Router1 must lower its path MTU estimate and fragment the Echo Request. TAR-Host1 must respond to the Echo Request sent by TAR-Router1 with an Echo Reply.

  Step 27: TAR-Host1 should send the Echo Request with fragmenting. TAR-Router1 must respond to the Echo Request sent by TAR-Host1 with an Echo Reply.
- *Part E*

  Step 30: TAR-Router1 should attempt to send the Echo Request without fragmenting. REF-Router1 must send an ICMPv6 Packet Too Big Message, and the TAR-Router1 must lower its path MTU estimate and fragment the Echo Request. TAR-Router2 must respond to the Echo Request sent by TAR-Router1 with an Echo Reply.

  Step 32: TAR-Router2 should send the Echo Request with fragmenting. TAR-Router1 must respond to the Echo Request sent by TAR-Router2 with an Echo Reply.

Possible Problems:

- A passive node may not implement an application for sending Echo Requests.
- Note: Some of these tests require the target device to configure link MTU on an interface. If this is not possible, the node may use a v6 over v4 tunnel (mtu = 1480), or a v6 over v6 tunnel (mtu = 1460). If the MTU is not configurable at all for the target device, this test may be omitted.

Test IP6Interop.1.7: Routing Header Processing

Purpose: Verify that devices can properly process a Routing header.

References:

- [IPv6-SPEC] – Section 4.4 and 8.4

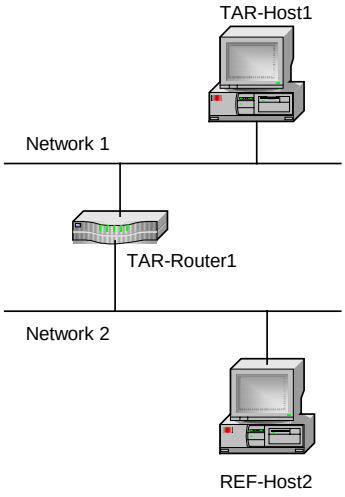
Resource Requirements:

- Monitor to capture packets
- Ref-Hosts must be able to transmit packets containing a Routing Header.

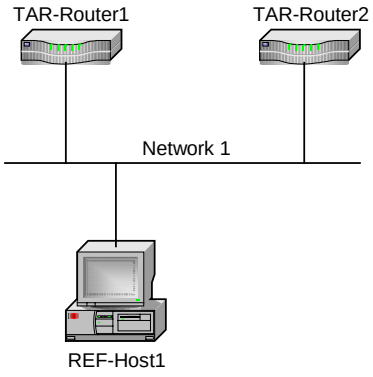
Discussion: The Routing header is used by an IPv6 source to list one or more intermediate nodes to be "visited" on the way to a packet's destination. When an upper-layer protocol sends one or more packets in response to a received packet that included a Routing header, the response packet(s) must not include a Routing header that was automatically derived by "reversing" the received Routing header unless the integrity and authenticity of the received Source Address and Routing header have been verified (e.g., via the use of an Authentication header in the received packet.)

Test Setup: Connect all devices as per the figures below. Allow time for all devices to perform stateless address autoconfiguration and Duplicate Address Detection.

Part A:



Part B:



Procedure:

Part A: Routing header (Host vs Router)

1. REF-Host2 transmits an ICMPv6 Echo Request with a Routing header. The Routing header is specified to go through TAR-Router1 and the destination TAR-Host1.
2. Observe the packets transmitted on Network1 and Network2.

Part B: Routing header (Router vs Router)

3. REF-Host1 transmits an ICMPv6 Echo Request with a Routing header. The Routing header is specified to go through TAR-Router1, then TAR-Router2 and then the destination REF-Host1.
4. Observe the packets transmitted on Network1.
5. REF-Host1 transmits an ICMPv6 Echo Request with a Routing header. The Routing header is specified to go through TAR-Router2, then TAR-Router1 and then the destination REF-Host1.
6. Observe the packets transmitted on Network1.
7. REF-Host1 transmits an ICMPv6 Echo Request with a Routing header. The Routing header is specified to go through TAR-Router1, then the destination TAR-Router2.
8. Observe the packets transmitted on Network1.
9. REF-Host1 transmits an ICMPv6 Echo Request with a Routing header. The Routing header is specified to go through TAR-Router2, then the destination TAR-Router1.
10. Observe the packets transmitted on Network1.

Observable Results:

- *Part A*

  Step 2: TAR-Host1 must transmit an Echo Reply to REF-Host2's Global Address using TAR-Router1 as a first hop. If the Echo Reply contains a Routing header, it must not be a reversal of the received Routing header.

- *Part B*

  Step 4: TAR-Router1 must receive the Echo Request from REF-Host1's address and forward the Echo Request to TAR-Router2 the next address in the Routing header. TAR-Router2 must receive the Echo Request from TAR-Router1 and forward the Echo Request to REF-Host1 the next address in the Routing header and the destination. REF-Host1 must receive the Echo Request from TAR-Router2.

  Step 6: TAR-Router2 must receive the Echo Request from REF-Host1's address and forward the Echo Request to TAR-Router1 the next address in the Routing header. TAR-Router1 must receive the Echo Request from TAR-Router2 and forward the Echo Request to REF-Host1 the next address in the Routing header and the destination. REF-Host1 must receive the Echo Request from TAR-Router1.

  Step 8: TAR-Router1 must receive the Echo Request from REF-Host1's address and forward the Echo Request to TAR-Router2 the next address in the Routing header. TAR-Router2 must receive the Echo Request from TAR-Router1 and transmit an Echo Reply to REF-Host1.

  Step 10: TAR-Router2 must receive the Echo Request from REF-Host1's address and forward the Echo Request to TAR-Router1 the next address in the Routing header. TAR-Router1 must receive the Echo Request from TAR-Router2 and transmit an Echo Reply to REF-Host1.

Possible Problems:

- None