

ROUTING CONSORTIUM TEST SUITE

Routing Information Protocol (RIP)
Over Internet Protocol Version 6

Technical Document

Version 2.0



**University of New Hampshire
InterOperability Laboratory
Routing Consortium
<http://www.iol.unh.edu>**

**121 Technology Drive, Suite 2
Durham, NH 03824
Phone: +1-603-862-2804
Fax: +1-603-862-0898**

Modification Record

*University of New Hampshire
InterOperability Laboratory*

ACKNOWLEDGMENTS

The University of New Hampshire would like to acknowledge the efforts of the following individuals in the development of this test suite.

Dr. William Lenharth	University of New Hampshire
Jeremy McCooey	University of New Hampshire
Sebastien Roy	University of New Hampshire
Quaizar Vohra	University of New Hampshire
Michael Briggs	University of New Hampshire
John Leser	University of New Hampshire
Ray LaRocca	University of New Hampshire
Kimo Johnson	University of New Hampshire
Brendan Libby	University of New Hampshire
Robert Wolff	University of New Hampshire
Catherine Rhoades	University of New Hampshire
Erica Williamsen	University of New Hampshire

INTRODUCTION

Overview

The University of New Hampshire's InterOperability Laboratory (IOL) is an institution designed to improve the interoperability of standards based products by providing an environment where a product can be tested against other implementations of a standard. This suite of tests has been developed to help implementers evaluate the functioning of their products supporting RIPng. The tests do not determine whether a product conforms to the RIPng specification, nor are they purely interoperability tests. Rather, they provide one method to isolate problems within a device. Successful completion of all tests contained in this suite does not guarantee that the tested device will operate with other RIPng devices. However, combined with satisfactory operation in the IOL's semi-production environment, these tests provide a reasonable level of confidence that the Router Under Test (RUT) will function well in most multi-vendor IPv6 environments.

Test Software

The UNH IOL Testing Software is not a complete implementation of IPv6 or RIPng; it is simply a packet generator that can transmit and receive packets. This allows the Testing Node to generate invalid packets and to simulate parts the RIPng protocol. All test routers and test nodes are used in these tests are simulated using the test software. The Testing Software is not currently available to the public.

Acronyms

RUT: Router Under Test
TN: Testing Node
TR: Testing Router
RTE: Route Table Entry

When several entities of the same type are present in a test configuration, a number is appended to the acronym to yield a label for each entity. For example, if there were three testing routers in the test configuration, they would be labeled TR1, TR2 and TR3.

TEST ORGANIZATION

This document organizes tests by group 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 group number and the test number within the group. These elements are separated by periods. The **Test Number** is the 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:

[RIPng] G. Malkin, R. Minnear, RIPng for IPv6, RFC 2080, January 1997

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

TABLE OF CONTENTS

MODIFICATION RECORD	2
ACKNOWLEDGMENTS	3
INTRODUCTION	4
TEST ORGANIZATION	5
REFERENCES	6
TABLE OF CONTENTS	7
GROUP 1 – PROCESSING	8
TEST RIPNG.1.1: BASIC RESPONSE PROCESSING	9
TEST RIPNG.1.2: NEXT HOP PROCESSING	10
TEST RIPNG.1.3: NEXT HOP OFF-LINK	11
TEST RIPNG.1.4: DEFAULT ROUTE PROCESSING	12
TEST RIPNG.1.5: EMPTY MESSAGES	13
TEST RIPNG.1.6: FULL TABLE REQUEST PROCESSING	14
TEST RIPNG.1.7: EQUAL METRIC ROUTES	15
TEST RIPNG.1.8: TRIGGERED RESPONSE GENERATION	16
TEST RIPNG.1.9: LARGE ROUTING TABLE RESPONSE GENERATION	17
TEST RIPNG.1.10: VERSION NUMBER FORWARD COMPATIBILITY	18
TEST RIPNG.1.11: NEIGHBOR LIST FUNCTIONALITY	19
TEST RIPNG.1.12: PREFIX LIST FUNCTIONALITY	20
GROUP 2 – INPUT VALIDATION	21
TEST RIPNG.2.1: MUST BE ZERO FIELDS	22
TEST RIPNG.2.2: INCORRECT UDP PORTS	23
TEST RIPNG.2.3: INCORRECT HOP COUNT	24
TEST RIPNG.2.4: INVALID ROUTE ENTRIES	25
TEST RIPNG.2.5: RESPONSE FROM OFF-LINK	26
TEST RIPNG.2.6: RESPONSE RECEIVED FROM ROUTER’S OWN ADDRESS	27
GROUP 3 – TIMERS	28
TEST RIPNG.3.1: ROUTE TIMEOUT AND GARBAGE COLLECTION	29
TEST RIPNG.3.2: ROUTE TIMER HALF EXPIRED HEURISTIC	31
TEST RIPNG.3.3: TRIGGERED RESPONSE DELAY INTERVAL	32
GROUP 4 – FORWARDING	33
TEST RIPNG.4.1: ROUTE PRIORITY	34

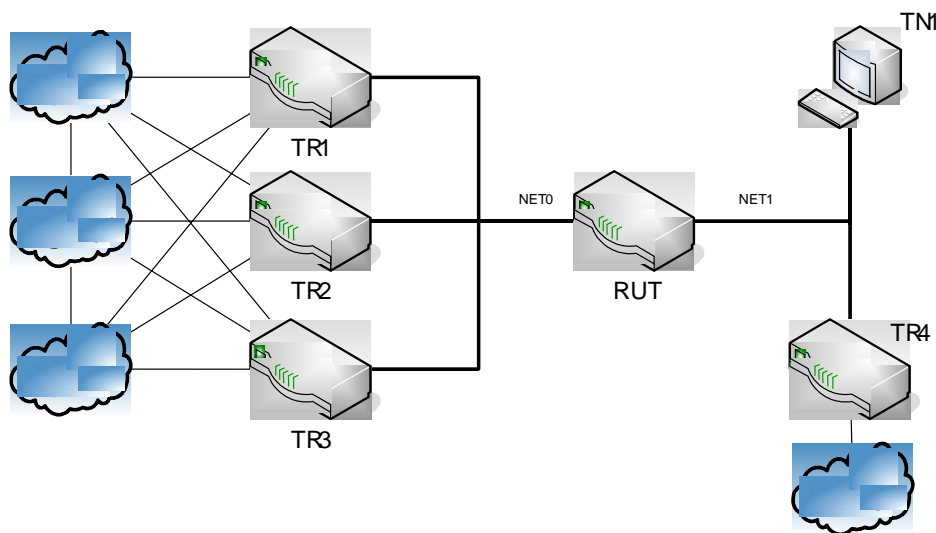
Group 1 – Processing

Overview

These tests are designed to verify the proper operation of a router running the routing protocol specified in [RIPng]. Tests in this group verify the capability of a RIPng enabled router to process valid routing messages and correctly propagate routing information.

Group 1 Common Test Setup and Cleanup:

The various routers and networks referred to in the course of the test descriptions are connected as follows:



For the purposes of testing, any of the test routers TR1 through TR3 may be connected to any networks, N3 through N100. These ‘connections’ are implied through the language of the test procedure; whenever a test router advertises a route to particular network, it will behave as though it were connected to that network. Networks labeled 3 and higher will have different network numbers in each test.

At the beginning of each test, the RUT is configured to run RIPng on its interfaces on N1 and N2. These interfaces are connected to two interfaces on the station to be used for packet generation and monitoring. Split-horizon processing with poisoned-reverse is enabled on the RUT, if available.

After each test is complete, the appropriate test routers generate RIPng messages with all metrics set to 16, in order to trigger route expiration for all routes which might have been learned by the RUT in the course of the test.

Test RIPNG.1.1: Basic Response Processing

Purpose: Verify that a router correctly processes a valid RIPng response and adds the routes advertised in the response to its routing table.

References:

- RFC 2080 – Section 2

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: Upon receiving a valid RIPng response, a router should process the route entries in the response one by one. New routes should be added to the routing table with the specified next hop. The route tags and prefix lengths associated with each route should be copied from the response message; the metrics should be incremented and included with the routes. The router should then send out a triggered response containing the newly added routes on all of its active RIPng interfaces, preserving the route tag and prefix length fields included with the routes in the original response. Depending on the configuration of the router, split horizon or reverse poisoning processing should be performed on the routes to be included in the response.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits a RIPng response message containing routes to networks N3 through N5. Each route has a different route tag.
2. Observe packets transmitted by the NUT.

Observable Results:

- After TR1 transmits the response message, the RUT should send a triggered response on N2 (note the split horizon configuration of the RUT). All of the routes learned from TR1 should be included with the correct metrics. The RUT should propagate the original route tag, as generated by TR1, with each route.

Possible Problems: None.

Test RIPNG.1.2: Next Hop Processing

Purpose: Verify that a router properly processes a RIPng response that contains multiple next hop entries.

References:

- RFC 2080 – Section 2

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: [RIPng] describes the next hop field as follows:

RIPng provides the ability to specify the immediate next hop IPv6 address to which packets to a destination specified by a route table entry (RTE) should be forwarded in much the same way as RIP-2 [2]. In RIP-2, each route table entry has a next hop field. Including a next hop field for each RTE in RIPng would nearly double the size of the RTE. Therefore, in RIPng, the next hop is specified by a special RTE and applies to all of the address RTEs following the next hop RTE until the end of the message or until another next hop RTE is encountered. A next hop is identified by a value of 0xFF in the metric field of the RTE.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits a RIPng response to the RUT. The response contains the following entries: Next Hop Entry for TR2, Route to N3, Next Hop Entry for ::, Route to N4.
2. TN1 transmits an echo request to the RUT destined for a node on each network (N3 and N4).
3. Observe the packets generated by the RUT.

Observable Results:

- The RUT should learn the routes advertised by TR1 with the given next hops. The RUT should forward the Echo Requests from TN1 as follows: the Echo Request destined for N3 should be sent to the link-layer address of TR2; the Echo Request destined to N4 should be sent to the link-layer address of TR1.

Possible Problems: None.

Test RIPNG.1.3: Next Hop Off-Link

Purpose: Verify that the RUT properly interprets an off link next hop router given in a RIPng response message as 0:0:0:0:0:0:0, the originator of the response.

References:

- RFC 2080 – Section 2.1

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: If the received next hop address is not a link-local address, it should be treated as 0:0:0:0:0:0:0.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits a RIPng response containing the following routes: Next Hop Entry(Global Address) of a TN on N4 (a network to which the RUT is not directly attached); a Route to N3.
2. TN1 transmits an Echo Request to the RUT destined to a node on N3.
3. Observe the packets generated by the RUT.

Observable Results:

- The RUT should consider TR1 as the next hop for the route to N3. Upon reception of the RIPng response in Step 1, the RUT should transmit a triggered response on N2, propagating a route to N3. The RUT should forward the Echo Request sent in Step 2 to TR1.

Possible Problems: None.

Test RIPNG.1.4: Default Route Processing

Purpose: Verify that the RUT properly adds default routes learned from received responses.

References:

- RFC 2080 – Section 2.2

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: A default route is advertised in a RIPng response packet by including a route entry that has both the prefix and the prefix length set to zero. The specification defines the default route as follows: Any prefix with a prefix length of zero is used to designate a default route. It is suggested that the prefix 0:0:0:0:0:0:0:0 be used when specifying the default route, though the prefix is essentially ignored. A default route is used when it is not convenient to list every possible network in the RIPng updates, and when one or more routers in the system are prepared to handle traffic to the networks that are not explicitly listed. These "default routers" use the default route as a path for all datagrams for which they have no explicit route.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits a RIPng response with a route entry of "::/0" indicating that it should be used as the default route.
2. TN1 transmits an Echo Request to the RUT destined for a node on N3.
3. Observe the packets generated by the RUT.

Observable Results:

- Upon receipt of the RIPng response from TR1 in Step 1, the RUT should transmit a triggered response on N2, propagating the default route. The RUT should forward the Echo Request sent in Step 2 to TR1.

Possible Problems: None.

Test RIPNG.1.5: Empty Messages

Purpose: Verify that the router does not produce any response on receipt of an empty request message, and that a router ignores an empty response message.

References:

- RFC 2080 – Section 2.4

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: A router should not generate any response upon receipt of a RIPng request or response that does not contain any entries.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

Part A. Empty Response

1. TR1 transmits a RIPng response containing no entries.
2. Observe the packets generated by the RUT.

Part B. Empty Request

3. TR1 transmits a RIPng request containing no entries.
4. Observe the packets generated by the RUT.

Observable Results:

- In Parts A and B, the RUT should not transmit any triggered responses. The RUT should ignore the empty messages.

Possible Problems: None.

Test RIPNG.1.6: Full Table Request Processing

Purpose: Verify that a router properly responds to a full table request.

References:

- RFC 2080 – Section 2.4

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: If there is exactly one entry in the request, and it has a destination prefix of zero, a prefix length of zero, and a metric of infinity (i.e., 16), then this is a request to send the entire routing table. In that case, a call is made to the output process to send the routing table to the requesting address/port. Note that there is a difference in metric handling for specific and whole-table requests. If the request is for a complete routing table, normal output processing is done, including Split Horizon.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits a RIPng response containing routes to N3 and N4.
2. TR4 transmits a RIPng full table request to the RUT.
3. TR1 transmits a RIPng full table request to the RUT
4. Observe the packets generated by the RUT.

Observable Results:

- The RUT should transmit a response to TR4 and TR1 that contains all of the route table entries advertised by TR1. If poison reverse processing is enabled, the RUT should set the metrics of routes to N3 and N4 to 16 when responding to TR1, since these routes were learned on that network.

Possible Problems: None.

Test RIPNG.1.7: Equal Metric Routes

Purpose: Verify that the router does not switch back and forth between multiple next hops for a route advertised at the same metric by more than one neighbor.

References:

- RFC 2080 – Section 2.4

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: Only in certain circumstances should a router choose a new next hop upon receiving an update containing a route already in its tables at the same metric. If the new metric is the same as the old one, it is simplest to do nothing further (beyond reinitializing the timeout); but, there is a heuristic which could be applied. Normally, it is senseless to replace a route if the new route has the same metric as the existing route; this would cause the route to bounce back and forth, which would generate an intolerable number of triggered updates. However, if the existing route is showing signs of timing out, it may be better to switch to an equally good alternative route immediately, rather than waiting for the timeout to happen. Therefore, if the new metric is the same as the old one, examine the timeout for the existing route. If it is at least halfway to the expiration point, switch to the new route. This heuristic is optional, but highly recommended.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits a RIPng response containing routes to N3 and N4 with a metric of 2.
2. TR4 transmits a RIPng response containing a route to N3 with a metric of 2.
3. TR1 transmits a RIPng response containing routes to N3 and N4 with a metric of 2.
4. TR4 transmits a RIPng response containing a route to N3 with a metric of 2.
5. TR1 transmits a RIPng response containing routes to N3 and N4 with a metric of 2.
6. TR4 transmits a RIPng response containing a route to N3 with a metric of 2.
7. TR1 transmits a RIPng response containing routes to N3 and N4 with a metric of 2.
8. TR4 transmits a RIPng response containing a route to N3 with a metric of 2.

Observable Results:

- Since TR4 never advertises a metric lower than TR1, the RUT should maintain TR1 as the next hop for N3 and not generate any triggered responses after the initial response.

Possible Problems: A router might keep both route entries in order to perform load balancing. In order to fail this test, a router must retain only one route at a time, and alternate between TR1 and TR4 as the next hop for that route. It will be necessary to check the routing table to verify this.

Test RIPNG.1.8: Triggered Response Generation

Purpose: Verify that the router includes only those routes that have changed when sending a triggered response.

References:

- RFC 2080 – Section 2.5

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: In order to minimize routing traffic, a triggered response should contain only those routes that have changed since that last response was sent. Therefore messages generated as part of a triggered update must include at least those routes that have their route change flag set. They may include additional routes, at the discretion of the implementor; however, sending complete routing updates is strongly discouraged.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. Wait for a periodic RIPng response from the RUT.
2. TR1 transmits a RIPng response with routes to N3, N4, N5, N6 and N7 at a metric of 3.
3. Wait 5 seconds.
4. TR1 transmits a RIPng response with routes to N5, N6 and N7 at a metric of 2.
5. Observe the packets generated by the RUT.

Observable Results:

- The RUT should send a triggered response on N2 upon receiving the second response message from TR1. This response should only contain routes to N5, N6 and N7 at the newly learned metric. If split-horizon processing is available and configured on the RUT, then the RUT should not generate any triggered response onto the N1 as a result of either of the updates sent by TR1.

Possible Problems: None.

Test RIPNG.1.9: Large Routing Table Response Generation

Purpose: Verify that the router sends more than one response when it would be impossible to include all of its routes in a single update due to the MTU of the link.

References:

- RFC 2080 – Section 2.5

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: When there are so many routes that sending them all in a single update would require that the update packet be over MTU, a router should send multiple update packets, each including a subset of the known routes.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits two responses that, when processed, result in a routing table larger than could be included in a single RIPng packet given the links MTU.
2. Observe the resulting packets.

Observable Results:

- Each time the RUT generates RIPng responses onto N2, it should correctly divide the contents of its routing table into multiple packets for transmission onto the link. The RUT's entire routing table should be included in each set of updates, with no duplicates or exclusions.

Possible Problems: None.

Test RIPNG.1.10: Version Number Forward Compatibility

Purpose: Verify that the router processes a RIPng packet with a numerically higher RIPng version number.

Reference:

- RFC 2080 – Section 2.5.2

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: To ensure that the router will be compatible with future versions of RIPng, a router should correctly process future version.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 sends a RIPng response with version number set to 2 and a route to N3.
2. Observe the packets sent by the RUT.

Observable Results:

- The RUT does not crash or generate invalid packets.

Possible Problems: None.

Test RIPNG.1.11: Neighbor List Functionality

Purpose: Verify that a router restricts the routers from which updates will be accepted when configured to do so.

Reference:

- RFC 2080 – Section 3.1.1

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: A neighbor list allows the network administrator to be able to define a list of neighbors for each router. A router would accept response messages only from routers on its list of neighbors. This functionality is optional.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. Configure the router to accept response messages from TR1 only.
2. TR1 sends a RIPng response with a route to N3.
3. TR4 sends a RIPng response with a route to N4.
4. Observe the packets transmitted by the RUT.

Observable Results:

- The RUT should not learn a route to N4 through TR4. It should learn the route to N3 through TR1. The route to N3 should be present in the RUT's responses transmitted on N2; a route to N4 should not be propagated.

Possible Problems: The RUT may not support the Neighbor List function.

Test RIPNG.1.12: Prefix List Functionality

Purpose: Verify that a router restricts the networks that it learns from response messages when configured to do so.

Reference:

- RFC 2080 – Section 3

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: A filter for specific destinations would permit the network administrator to be able to specify a list of destination prefixes to allow or disallow. If configured to disallow learning certain prefixes, the RUT must ignore RTEs for that prefix contained in RIPng responses.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. Configure the RUT to disallow learning the prefix of N3.
2. Configure the RUT to disallow learning routes to an aggregated (48 bit) prefix including N4.
3. TR1 sends a RIPng response that contains routes to N3, N4 and N5.
4. Observe the packets sent by the RUT.

Observable Results:

- The RUT should not learn the routes to N3 or N4, but should learn the route to N5. Only the route to N5 should be propagated in the RUT's responses sent on N2.

Possible Problems: The RUT may not support the Prefix List function.

Group 2 – Input Validation

Overview

These tests verify the capability of a RIPng enabled router to validate incoming RIPng packets. The various checks mandated for incoming packets ensure that a router doesn't process possibly corrupt routing information.

Group 2 Common Test Setup and Cleanup:

These tests use the same setup provided for the Group 1 tests.

Test RIPNG.2.1: Must be Zero Fields

Purpose: Verify that the RUT ignores data present in the “must be zero” portions of next hop fields of a RIPng response.

References:

- RFC 2080 – Section 2.1

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: Certain fields of a next hop RTE in the RIPng response go unused and should be set to zero. The route tag and prefix length in the next hop RTE must be set to zero on sending and ignored on reception. The fields are also referred to as the “must be zero” or mbz fields in the specification.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits a response with the following routes: Next Hop Entry for TR2; Route to N3, Next Hop Entry for ::; Route to N4. The second next hop entry has non-zero values in the mbz fields.
2. TN1 transmits an Echo Request to a node on N4.
3. Observe the packets generated by the RUT.

Observable Results:

- The RUT should forward the Echo Request destined to the node on N4 to the hardware address of TR1.

Possible Problems: None.

Test RIPNG.2.2: Incorrect UDP Ports

Purpose: Verify that the router ignores responses received on or sent from any UDP port other than the designated RIPng port.

References:

- RFC 2080 – Section 2.1

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Last Modification: October 17, 1997

Discussion: The UDP port 521 has been designated as the port to use for RIPng traffic. All routing update messages should be sent to and from this port. A router should ignore any RIPng responses received on any port other than 521, and should ignore any response indicating a source UDP port other than 521.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

Part A: Incorrect Source Port

1. TR1 transmits a RIPng response from source UDP port other than 521, with a route to N3
2. Observe the packets generated by the RUT.

Part B: Incorrect Destination Port

3. TR1 transmits a RIPng response to a destination UDP port other than 521, with a route to N4.
4. Observe the packets generated by the RUT.

Observable Results:

- In Parts A and B, the RUT should ignore both responses; routes to N3 or N4 should not be present in the RUT's responses on N2.

Possible Problems: None.

Test RIPNG.2.3: Incorrect Hop Count

Purpose: Verify that a router ignores RIPng responses that have an IPv6 hop count less than 255.

References:

- RFC 2080 – Section 2.4.2

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: Inbound, multicast packets sent from the RIPng port (i.e. periodic advertisement or triggered update packets) must be examined to ensure that the hop count is 255. This absolutely guarantees that a packet is from a neighbor, because any intermediate node would have decremented the hop count.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits a RIPng response with a hop limit less than 255, with a route to N3.
2. Observe the packets generated by the RUT.

Observable Results:

- The RUT should not learn the route to N3 from the RUT; the RUT should not include a route to N3 in its responses sent on N2.

Possible Problems: None.

Test RIPNG.2.4: Invalid Route Entries

Purpose: Verify that a router uses the proper criteria in order to validate the route entries in a route update message, and does not add or propagate invalid routes.

References:

- RFC 2080 – Section 2.4

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: A number of criteria must be met before a router should copy the routes from a route update into its routing table for use:

- Is the destination prefix valid (e.g., not a multicast prefix and not a link-local address) A link-local address should never be present in an RTE.
- Is the prefix length valid (i.e., between 0 and 128, inclusive)
- Is the metric valid (i.e., between 1 and 16, inclusive)

If any check fails, ignore that entry and proceed to the next.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits a RIPng response with 5 route table entries. The first route table entry is a multicast prefix, the second is a link-local prefix, the third has a prefix length of 129, the fourth has a metric of 17, and the fifth is a valid route entry to N3.
2. Observe the packets generated by the RUT.

Observable Results:

- Only the route to N3 should be included in the triggered and periodic responses from the RUT. None of the invalid routes should be present in the RUT's response messages on N2 or in the RUT's routing table.

Possible Problems: None.

Test RIPNG.2.5: Response from Off-Link

Purpose: Verify that the router ignores a route update sent from an off link, global address.

References:

- RFC 2080 – Section 2.4

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: The datagram's IPv6 source address should be checked to see whether the datagram is from a valid neighbor; the source of the datagram must be a link-local address.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits a RIPng response onto N1, with a route to N3; the source IP address of this packet is TR1's global IPv6 address on N4.
2. Observe the packets generated by the RUT.

Observable Results:

- The RUT should not learn the route to N3. The RUT should not include a route to N3 in its responses sent on N2.

Possible Problems: None.

Test RIPNG.2.6: Response Received from Router's Own Address

Purpose: Verify that the router ignores RIPng responses from its own address.

Reference:

- RFC 2080 – Section 2.4.2

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: It is worth checking to see whether a response is from one of the router's own addresses. Interfaces on broadcast networks may receive copies of their own multicasts immediately. If a router processes its own output as new input, confusion is likely, and such datagrams must be ignored.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits a RIPng response onto N1 with a source address equal to the IPv6 address of the RUT on N1. This response contains a route to N3.
2. Observe the packets generated by the RUT.

Observable Results:

- The RUT should not learn the route to N3. The RUT should not include a route to N3 in its responses sent on N2.

Possible Problems: None.

Group 3 – Timers

Overview

These tests verify that a RIPng enabled router properly implements protocol timers specified in [RIPng].

Group 3 Common Test Setup and Cleanup:

These tests use the same setup provided for the Group 1 tests.

Test RIPNG.3.1: Route Timeout and Garbage Collection

Purpose: Verify that the router properly triggers and handles route expiration and garbage collection.

References:

- RFC 2080 – Section 2.3

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: Deletions can occur for one of two reasons: the timeout expires, or the metric is set to 16 because of an update received from the current router.

There are two timers associated with each route, a "timeout" and a "garbage-collection time." Upon expiration of the timeout, the route is no longer valid; however, it is retained in the routing table for a short time so neighbors can be notified that the route has been dropped. Upon expiration of the garbage-collection timer, the route is finally removed from the routing table.

The timeout timer is set for 180 seconds. The garbage collection timer is set for 120 seconds after timeout occurs.

Until the garbage-collection timer expires, the route is included in all updates sent by this router. When the garbage-collection timer expires, the route is deleted from the routing table. When a route is included in response during garbage collection, its metric is set to 16 (infinity).

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of Part A. In addition, a common cleanup procedure is performed after Part D.

Procedure:

1. TR1 transmits a RIPng response with routes to N3, N4 and N5, all with metrics of 1.
2. Wait 15 seconds.
3. TR1 transmits the same RIPng response, with the metric for the route to N3 set to 16.
4. Observe the packets generated by the RUT.
5. Wait 150 seconds.
6. Observe the packets generated by the RUT.
7. TR1 transmits a response containing only a route to N5, with metric of 1.
8. Wait 30 seconds.
9. TR2 transmits a RIPng full table route request to the RUT.
10. Observe the packets generated by the RUT.
11. Wait 155 seconds.
12. Observe the packets generated by the RUT.

Observable Results:

- **Step 4:** The RUT should immediately transmit a triggered RIPng update onto N2 upon receipt of the response from TR1 expiring the route to N3. This response should include a route to N3 with metric 16.

*University of New Hampshire
InterOperability Laboratory*

- **Step 6:** The RUT should include the route to N3 in its periodic updates until 120 seconds elapse from when the route expired (in Step 3).
- **Step 10:** 180 seconds after the RUT learned the route to N4, in Step 1, the route to N4 should expire. The RUT should generate a triggered response onto N2 including a route to N4 with metric 16. Also the RUT should send to TR2, a full routing response.
- **Step 12:** 180 seconds after the RUT updated the route to N5, in Step 7, the route to N5 should expire. The RUT should generate a triggered response onto N2 including a route to N5 with metric 16.

Possible Problems: None.

Test RIPNG.3.2: Route Timer Half Expired Heuristic

Purpose: Verify that the router switches from one next hop to another with the same metric if the current timer is at least half expired.

References:

- RFC 2080 – Section 2.4.2

Resource Requirements

- Packet Generator
- Monitor to capture packets

Discussion: If the new metric is the same as the old one, examine the timeout for the existing route. If it is at least halfway to the expiration point, switch to the new route. This heuristic is optional, but highly recommended.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits a RIPng response with a route to N3, metric 1.
2. Wait 110 seconds
3. TR4 transmits a RIPng response with a route to N3, metric 1.
4. Observe the packets generated by the RUT.

Observable Results:

- Because the timer for its route to N3 is more than half expired when the response from TR4 is received in Step 3, the RUT should adopt the route to N3 through TR4. This can be seen through the transmission of triggered updates on N1.

Possible Problems: A router is not required to employ this heuristic.

Test RIPNG.3.3: Triggered Response Delay Interval

Purpose: Verify that the router waits the proper amount of time between sending triggered responses.

References:

- RFC 2080 – Section 2.5

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: In order to keep the triggered updates on a link from becoming synchronized, a random interval is introduced in the delay between each update. After a triggered update is sent, a timer should be set for a random interval between 1 and 5 seconds. If other changes that would trigger updates occur before the timer expires, a single update is triggered when the timer expires. The timer is then reset to another random value between 1 and 5 seconds. Triggered updates may be suppressed if a regular update is due by the time the triggered update would be sent.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part.

Procedure:

1. TR1 transmits a RIPng response with a route to N3, metric 8.
2. Wait 500 milliseconds.
3. Repeat Steps 1 and 2 six additional times, each time reducing the metric of the route to N3 by 1.
4. Observe the packets generated by the RUT.

Observable Results:

- While each packet from TR1 would normally cause the RUT to generate a triggered response on N2. The RUT should delay a random interval of 1 – 5 seconds between triggered responses.

Possible Problems: None

Group 4 – Forwarding

Overview

These tests verify the capability of a router to forward packets based on a routing table populated by the RIPng protocol.

Group 4 Common Test Setup and Cleanup:

These tests use the same setup provided for the Group 1 tests.

Test RIPNG.4.1: Route Priority

Purpose: Verify that the proper routes are used when more than one route is available for a given network.

References:

- RFC 2080

Resource Requirements:

- Packet Generator
- Monitor to capture packets

Discussion: Whenever a router must choose between multiple routes that all match a given prefix, it must choose the most specific route first. Whichever route has the longest match with the target prefix should be used. A more specific route should be chosen even though another route is available with a lower metric. However, if multiple routes have the same level of specificity, then the one with the lowest metric should be used.

Test Setup: Refer to the common test setup for Group 1 tests. The common setup is performed at the beginning of each test part. In addition, a common cleanup procedure is performed after each part. Ensure that N3's network prefix length is 64-bits.

Procedure:

Part A: Default Route

1. TR1 transmits a RIPng response with a single default route entry, metric 3.
2. TN1 sends an Echo Request message to the RUT destined for a node on N3.

Part B: Longer Prefix Match (48-bit prefix)

3. TR1 transmits a RIPng response with a single default route entry, metric 3.
4. TR2 transmits a RIPng response with a route to the 48-bit prefix that includes N3, metric 4.
5. TN1 sends an Echo Request message to the RUT destined for a (different) node on N3.

Part C: Better Metric (48-bit prefix)

6. TR1 transmits a RIPng response with a single default route entry, metric 3.
7. TR2 transmits a RIPng response with a route to the 48-bit prefix that includes N3, metric 4.
8. TR1 transmits a RIPng response with a route to the 48-bit prefix that includes N3, metric 2.
9. TN1 sends an Echo Request message to the RUT destined for a (different) node on N3.

Part D: Longer Prefix Match (64-bit prefix)

10. TR1 transmits a RIPng response with a single default route entry, metric 3.
11. TR2 transmits a RIPng response with a route to the 48-bit prefix that includes N3, metric 4.
12. TR1 transmits a RIPng response with a route to the 48-bit prefix that includes N3, metric 2.
13. TR2 transmits a RIPng response with a route to N3, metric 5, and prefix length of 64.
14. TN1 sends an Echo Request message to the RUT destined for a (different) node on N3.

Part E: Better Metric (64-bit prefix)

15. TR1 transmits a RIPng response with a single default route entry, metric 3.
16. TR2 transmits a RIPng response with a route to the 48-bit prefix that includes N3, metric 4.

*University of New Hampshire
InterOperability Laboratory*

17. TR1 transmits a RIPng response with a route to the 48-bit prefix that includes N3, metric 2.
18. TR2 transmits a RIPng response with a route to N3, metric 5, and prefix length of 64.
19. TR1 transmits a RIPng response with a route to N3, metric 1.
20. TN1 sends an Echo Request message to the RUT destined for a (different) node on N3.

Observable results:

- In Part A, the RUT should forward the Echo Request destined for N3 to the link-layer address of TR1.
- In Part B, the RUT should forward the Echo Request destined for N3 to the link-layer address of TR2.
- In Part C, the RUT should forward the Echo Request destined for N3 to the link-layer address of TR1.
- In Part D, the RUT should forward the Echo Request destined for N3 to the link-layer address of TR2.
- In Part E, the RUT should forward the Echo Request destined for N3 to the link-layer address of TR1.

Possible Problems: None.